September '84

Das Magazin für den Home-Computer-Besitzer

computronic

9/1. Jahrgang

DM 5,50
öS 45
SFR 5,50



Programm-Hits

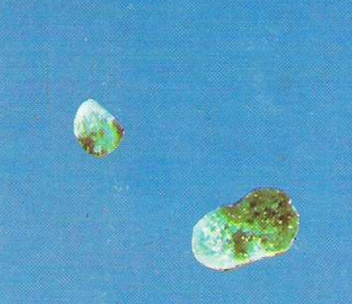
ausgewählt von der Redaktion:

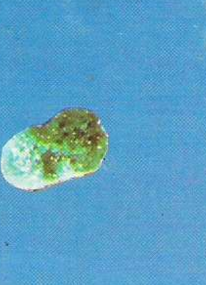
Projekt
Jump About

sowie

Silver-Spar
Crazy-Cake
Fünf gewinnt
The Big Quest
Schloß Gruselstein

und viele mehr









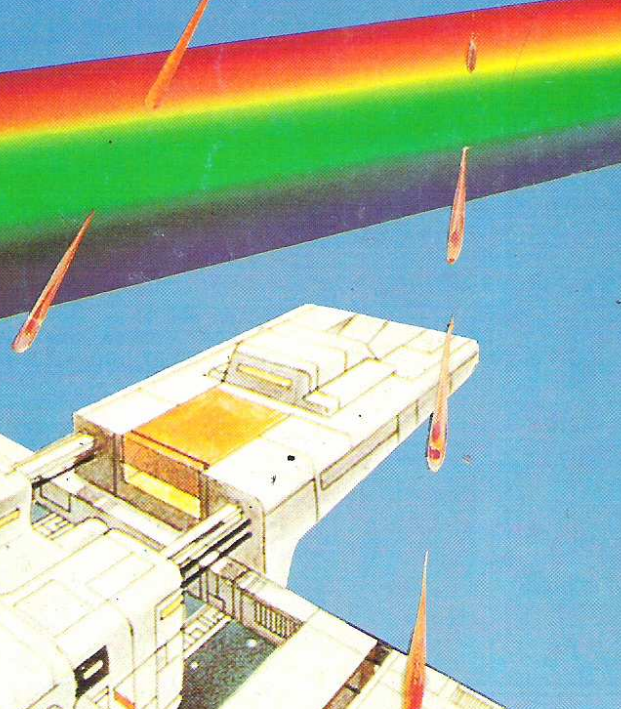
Anwenderprogramme

Forth-Kurs, Teil 6

Programmschleifen

Software Reviews

VC-20
TI-99
Apple II
ZX-81
Dragon 32
Atari
Laser 2001
Colour Genie
Commodore 64
ZX-Spectrum



Möchten Sie als Software-Autor für den TRONIC-Verlag tätig werden?

Wir würden uns freuen, in Ihnen einen Ansprechpartner für eine künftige intensive Zusammenarbeit zu finden. Der TRONIC-Verlag vereinbart mit seinen künftigen Software-Autoren pro veröffentlichte volle Seite (einschl. Programmbeschreibung) ein Honorar von **DM 120,–.**

Dieses Entgelt wird fällig, wenn die Redaktion des Verlages sich für eine Veröffentlichung entscheidet. Die Auszahlung erfolgt also nicht erst nach Veröffentlichung in einer unserer Ausgaben, sondern früher.

Der Verlag wird vom Autor berechtigt, seine Manuskripte (Programme) zur Darstellung im Heft heranzuziehen und abzudrucken.

Einzusenden sind:

- Programmbeschreibung
- bespielte Cassette oder Diskette
- Listing (mit Copyright)
- Freiumschlag

**Rufen Sie uns an, sollten Sie ★ Verbesserungsvorschläge ★ gute Programme ★ Tips & Tricks oder andere gute ★ Ideen zur Vervollständigung unseres Angebotes in »Computronic« haben. Wobei wir natürlich auch für jede ★ kritische Anmerkung dankbar sind.
Wir melden uns bestimmt!
0 56 51 / 4 06 93
oder
0 56 51 / 4 06 43**

Zusätzlich kann jeder Einsender eines Programmes eine kostenlose Kleinanzeige in Computronic zum Programm hinzufügen. Die, gleichgültig ob sein Programm veröffentlicht wird, in der nächsten Computronic erscheint!

Haben Sie Interesse? Haben Sie noch Fragen?

Setzen Sie sich telefonisch mit unserer Redaktion in Verbindung!

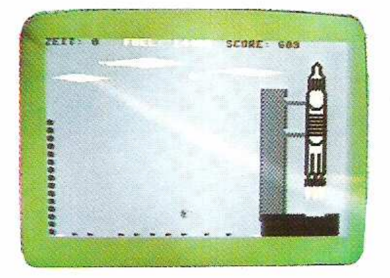
TRONIC-VERLAG
DIE REDAKTION

Aus dem Inhalt:



Wieder ein interessantes Spiel für den **Laser 2001**, bei dem Sie schnell reagieren müssen. Aber sehen Sie selbst!

– Project – für den C-64.



Ihre Aufgabe ist es, einen Flug ins Universum zu unternehmen, um dort Uran zu sammeln. Treten Sie nun, ausgerüstet mit Rakete, Mondfähre, Geigerzähler, Fallschirm usw. Ihre Reise an.

– Jump about – für den ZX-Spectrum.



Ein Spiel, das dem bekannten Jumpman an Schwierigkeitsgrad und interessanten Bildern in nichts nachsteht.

In diesem Heft:

Software

Aktuelles

In eigener Sache

Liebe Leser,

was uns während und nach der Erstellung der Hefte 6 + 7 so alles passiert ist, gehört sicherlich zum journalistischen Alltag.
Leidtragender, und dies schmeckt uns ganz und gar nicht, ist der Leser.
Die angesprochenen Hefte waren schlecht im Druck. Teilweise waren Passagen der Listings nicht lesbar. Unsere Leser waren sauer – wir auch!
Unser Service konnte daher nur darin liegen, wunschgemäß besser lesbare Listings in Fotokopie demjenigen Leser, der sie anforderte, zuzusenden. Wir hoffen, daß dies positiv aufgenommen wurde.
Uns fiel des weiteren auf, daß unser „Kassettenservice" für den TI 99 keine Kassetten und Disketten und für den Apple keine Disketten angeboten hat. Das Fehlen wurde bei der Korrektur übersehen – selbstverständlich könnt Ihr bestellen.

Dazu: In den Heften 8 und 9 haben wir unseren Kassettenservice neu geordnet und auch auf unseren „Blitz-Service" hingewiesen – POSTEINGANG = POSTAUSGANG. Schneller geht's wirklich nicht. Laßt Euch überraschen.
Und dann der Hammer mit den „geklauten" Programmen. So geht's nun wirklich nicht! Jeder Leser hat Anspruch auf neue, selbstentwickelte Programme. Unsere Programmierer jedenfalls bringen immer wieder neue und gute Software. Wir werden zukünftig solche Zuschriften nicht mehr beantworten.
Unsere Redaktion fragt – Hersteller oder Händler antworten nicht! Um eigene Programme und uns zugesandte Programme entwickeln und testen zu können, haben wir selbstverständlich jedes angebotene Computer-System im Hause.
Nun war unser DRAGON 32 defekt. Schnelles Handeln war notwendig. Wie wenig unsere (und letztlich auch Eure) Partner Verständnis zeigen, wurde uns am Beispiel der Fa. VOBIS deutlich. Ein Service, der offensichtlich bei diesem Händler klein geschrieben wird! Dazu kann man nur feststellen:
BESTELLEN gern – SERVICE nein!!!
Aber auch ein LOB: ATARI zeigt nach wie vor Kooperationsfähigkeit. Weiter so, wir werden dies zu würdigen wissen.
Euch liegt jetzt Heft 9 vor, wie gefällt es Euch? Seid Ihr von den Programmen begeistert? Habt Ihr Wünsche und Anregungen?
Schreibt uns, wir wollen Eure Wünsche berücksichtigen, damit Ihr auch weiterhin unsere treuen Leser bleibt.
Denn – wir sind anders als die anderen.

Bis demnächst
Eure Computronic-Redaktion

Schallschluckhaube

Drucken im Flüsterton ist keine Hexerei – mit Schallschluckhauben kann das Druckgeräusch vermindert werden. Das neue Schallschluckgehäuse des Versandhändlers Inmac soll einen Geräuschpegel von 70 dB auf unter 50 dB reduzieren. Das Acrylgehäuse spiegelt und reflektiert laut Anbieter nicht – die Sicht auf das bedruckte Papier soll eben unbeeinträchtigt bleiben.
Das lärmdämmende Gehäuse gibt es in drei Versionen – auch für Drucker mit Einzelblatteinzug. Eine Probezeit von 30 Tagen gehört zum Service der Firma.
Inmac GmbH
Frankfurter Straße 103
6096 Raunheim

Messetermine 84/85

6.– 8. Sept. Commodore Fachausstellung Frankfurt
17.–21. Okt. Hobby-Electronic Stuttgart
14.–18. Nov. Comdex Las Vegas
3.– 6. Dez. Comdex Amsterdam
13.–16. Juni 85 Internationale Computershow Köln

Fachausstellungen „Computertage 84"
31. Aug.– 2. Sept. Kaiserslautern
1. Sept.– 4. Sept. Göttingen
4. Sept.– 6. Sept. Gießen
14. Sept.–16. Sept. Hagen
22. Sept.–24. Sept. Duisburg
12. Okt. –14. Okt. Siegen
19. Okt. –21. Okt. Ludwigshafen
23. Nov.–25. Nov. Fulda
1. Dez. – 5. Dez. Heidelberg
8. Dez. – 9. Dez. Neu Isenburg
5. Hessische Computertage

Wayne Creekmore

Mikro-Wissen griffbereit

Wer seine Mitarbeiter auf den betrieblichen Mikro-Einsatz vorbereiten wollte, hatte bisher schon verschiedene Möglichkeiten, dies zu tun. Diese Palette hat sich nun um eine neue, kostengünstige Möglichkeit erweitert: den illustrierten Leitfaden „Mikro-Wissen griffbereit" von Creekmore.
Ohne sich mit langatmigen Betrachtungen über die Entwicklung der Rechentechnik vom Abakus bis zu den EDV-Großanlagen aufzuhalten, kommt der Autor ohne Umschweife direkt zur Sache. So erfährt der Leser bereits auf der ersten Textseite, daß ein Computer nichts anderes kann als bestimmte mathematische Operationen auszuführen, daß er nicht bei allen Aufgaben optimal eingesetzt ist, daß aber niemand schneller als er die oft zitierte „Nadel im Heuhaufen" findet. Die nächste Textseite erklärt dann an einem gut gewählten Beispiel den Unterschied zwischen Hardware und Software bzw. zwischen Hardware einerseits und Betriebssystem und Anwenderprogrammen andererseits.
In der Folge werden dann Seite für Seite wichtige Vorgänge beim professionellen und kommerziellen Einsatz erläutert. So gibt die Seite 16 beispielsweise an, wie die Grundbausteine Eingabe, CPU, interner Speicher, externer Speicher und Ausgabe zusammenwirken, wenn eine editierte Datei abgespeichert werden soll und was mit der Originaldatei geschieht.
Die komprimierte, durch eine aufwendige Graphik unterstützte Darstellung hält den Leser 64 Seiten lang bei der Stange. Am Ende weiß er dann das Wichtigste über die interne Datenverarbeitung, über Computersprachen, Anwender-Software, die Hardware-Ausstattung und die Planung eines Mikro-Kaufs. Dieses Wissen gibt Sicherheit bei der Lektüre weiterführender Bücher, bei Schulungsmaßnahmen und Seminaren und beim Computer-Einsatz selbst.
Braunschweig: Vieweg/Frankfurt: Ashton-Tate 1984, DM 24,80

Dateiverwaltung für Einsteiger

Damast nennt sich ein Dateienprogramm für den Home-Computer TI 99/4A von Texas Instruments. Es ist hauptsächlich geeignet zum Erstellen und Verwalten von Bestands-Dateien wie zum Beispiel Adressen-Karteien. Als Hardware ist der TI 99/4A mit Speichererweiterung und Diskettensystem erforderlich.
Laut Anbieter gehen die Fähigkeiten von Damast weit über die des TI-eigenen Moduls „Datenmanagement und Analyse" hinaus. So kann der Anwender den Aufbau seiner Datenbank, der Bildschirmmaske und der Druckformate frei wählen. Mit den Druckprogrammen lassen sich Adreßaufkleber, Tabellen und Listen erstellen.
Ein mitgeliefertes Textprogramm erlaubt das Mischen von Texten und Dateiinhalten, etwa zum Erstellen von Serienbriefen. In allen Druckprogrammen können die benötigten Datensätze frei selektiert werden. Das Programm ist aus mehreren Einzelmodulen zusammengesetzt, die je nach Bedarf miteinander kombiniert werden können.
Der Preis bewegt sich entsprechend zwischen 200 und 600 Mark.
Computer-Hüsli
Hippelstr. 37
8000 München 82

Games of the Olympic for Computer

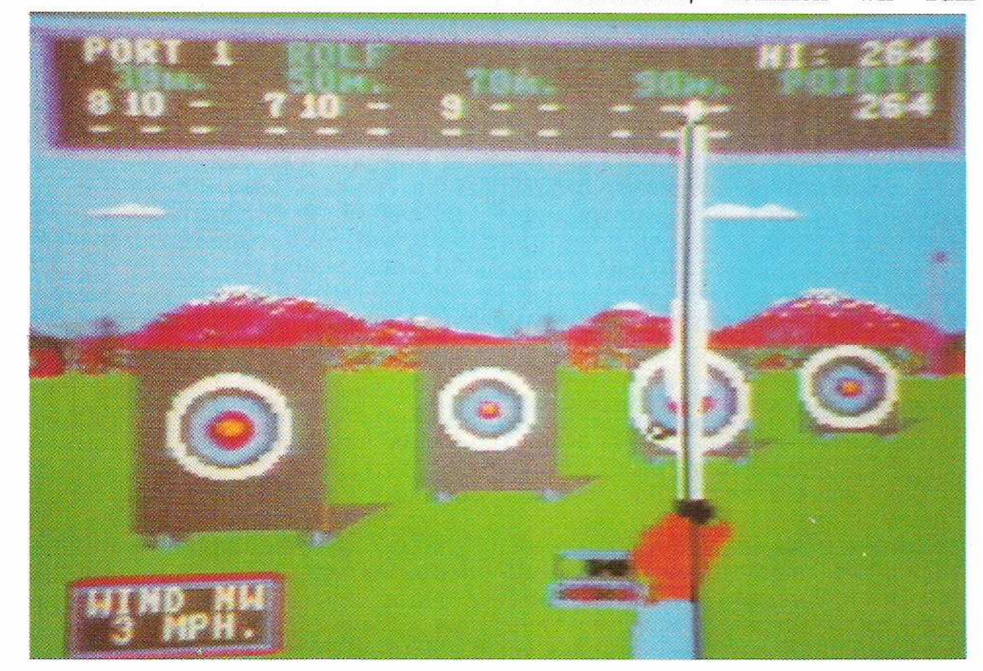
Nachdem die Olympischen Spiele in Los Angeles nun zu Ende sind, und wir Ringe unter den Augen haben, sind für den Computer neben Summer Games (wir haben das Spiel in der August-Ausgabe vorgestellt) jetzt auch Hes Games und Decathlon im Handel erhältlich. Nun können wir auch nach den Medaillen greifen.

Hes Games

Hes Games setzt sich aus 6 Sportarten zusammen, wie 100-Meter-Lauf, 110 Meter Hürden, Weitsprung, Bogenschießen, Sprung vom 1-Meter-Brett und Gewichtheben. Nachdem das Programm eingeladen ist, besteht die Möglichkeit, seinen Namen, die Nation, Shirtfarbe und Sockenfarbe einzugeben. Der Joystickport läßt sich durch Bewegung des Joysticks ändern. Im Menue besteht die Auswahlmöglichkeit zwischen den einzelnen Sportarten.
Bevor wir zum 100-Meter-Start kommen, kann man sich den Weltrekord auf der Anzeigentafel zeigen lassen. Nach Aufforderung zum Start muß der Läufer durch seitliche Bewegungen des Joysticks über die Laufbahn bewegt werden. Je schneller der Joystick bewegt wird, desto schneller der Läufer.
Beim 110-Meter-Hürdenlauf sind die gleichen Bedingungen wie beim 100-Meter-Lauf. Wir müssen beim Überspringen der Hürden mit den Bewegungen des Joysticks aufhören, da sonst der Läufer stolpert.
Um große Weiten im Weitsprung zu erzielen, muß der Absprung genau getroffen werden, beim Erreichen der Absprungmarke ist der Joystick nach oben zu drücken und bei der Landung nach rechts. Weiten von 8 Metern sind keine Seltenheit. Natürlich muß der Springer auch eine dementsprechende Geschwindigkeit haben.
Vom 1-Meter-Brett gekonnt ins kühle Naß einzutauchen, ohne auf den Bauch oder Rücken zu fallen, ist auch



nach längerer Zeit des Spielens sehr schwer. Traumnoten haben wir hier nicht erreicht. Aber Übung macht den Meister.
Sein Können muß man beim Bogenschießen beweisen. Zum Spannen des Bogens muß die Feuertaste gedrückt und bis zum Schuß gehalten werden. Durch ziehen des Joysticks nach hinten wird der Bogen gespannt. Nun kann man auf eine Distanz von 30, 50, 70 und 90 Meter seinen Pfeil ins Ziel bringen. Wobei der Wind, der aus allen Richtungen und mit wechselnder Stärke auftritt, berücksichtigt werden muß.
Wohl zu der schwersten Sportart, dem Gewichtheben, kommen wir zum Schluß. Zwei Arten von Gewichtheben sind zu absolvieren. Das Stoßen und das Reißen. Das Heben der Gewichte ist durch rechtzeitiges Vor- und Zurückbewegen des Joysticks eine schwere Sache.



Decathlon

Neben *Summer Games* und *Hes Games* ist *Decathlon* die dritte Sportart, die zum sportlichen Wettkampf aufruft.
Decathlon umfaßt 10 Sportarten, wie 100-Meter-Lauf, Weitsprung, Kugelstoßen, Hochsprung, 400-Meter-Lauf, Diskus, Hochsprung, Speerwurf und 1500-Meter-Lauf. Also ein echter Zehnkampf, den man mit vier Spielern spielen kann.
Besondere Ansprüche werden an die Spieler nicht gestellt, denn alle Sportarten werden durch seitliche Bewegungen des Joysticks ausgeführt. Zum Absprung ist die Feuertaste zu drücken.
Beim 100-Meter-Lauf, 110 Meter Hürden, 400-Meter-Lauf und 1500-Meter-Lauf spielen Einzelspieler gegen den Computer, der es jedem Spieler natürlich besonders schwer macht, zu gewinnen. Auf eine ausführliche Beschreibung des Programms wurde verzichtet, da das Programm vom Ablauf her *Summer Games* und *Hes Games* gleicht.

Zusammenfassung

Alle drei Spiele sind empfehlenswert, vor allem darum, weil diese Programme Kinder wie Eltern gleichermaßen begeistern. Eine Vergabe der Noten würde wie folgt aussehen: *Summer Games* „sehr gut", *Hes Games* „gut" und *Decathlon* „befriedigend". *Summer Games* wurde Testsieger, weil einfach alles stimmt, wie Grafik, Bewegungsablauf, Farben und vieles mehr. Alle Spiele gesehen bei: SOFTSELL in München.

Rolf Freitag

Kinder lernen am Computer

Ein von Erziehungswissenschaftlern geleiteter Modellversuch zeigt bisher: Kindern macht der Umgang mit Computern Spaß. In ihrer Freizeit ziehen sie jedoch den Sport dem Elektronenhirn vor.

Unter der Leitung von drei Erziehungswissenschaftlern der Universität in Saarbrücken hat in der Kindertagesstätte der saarländischen Stadt Ottweiler (Kreis Neukirchen) ein Modellversuch begonnen, bei dem zehn Jungen und Mädchen zwischen vier und vierzehn Jahren an einem rund 5000 Mark teuren Elektronengehirn lernen. Bisherige Zwischenbilanz: Den Kindern bereitet das Spielen am Computer durchweg Spaß und sie lernen dabei logisches Denken, doch fahren sie nach wie vor lieber Rad, Turnen oder spielen lieber Fußball.

„Komm her, ich zeig Dir das mal", sagt der vor dem Bildschirm, Diskette und Schreibmaschinentastatur sitzende vierjährige Jens, wenn in der Kindertagesstätte in Ottweiler mal wieder ein interessierter, aber computerunerfahrener Reporter auftaucht. Der gerade 1,08 Meter große Blondschopf, der noch Papa oder Mama braucht, wenn er „mal aufs Klo muß", steht mit dem Computer bereits auf Du und Du. Mit dem kleinen rechten Zeigefinger auf die richtige Taste drücken – und schon erscheint in grüner Schrift auf dem Computer-Bildschirm die Frage: „Wie heißt Du? Schreibe Deinen Namen und drücke die Abschick-Taste!" Vom sechsjährigen Rosario und dem neunjährigen Michael, der schon die Schule besucht, hat Jens mittlerweile gelernt, daß er nun wie auf der Schreibmaschine die Buchstaben J-e-n-s tippen und die Sendetaste drücken muß.

Gegenseitige Hilfe

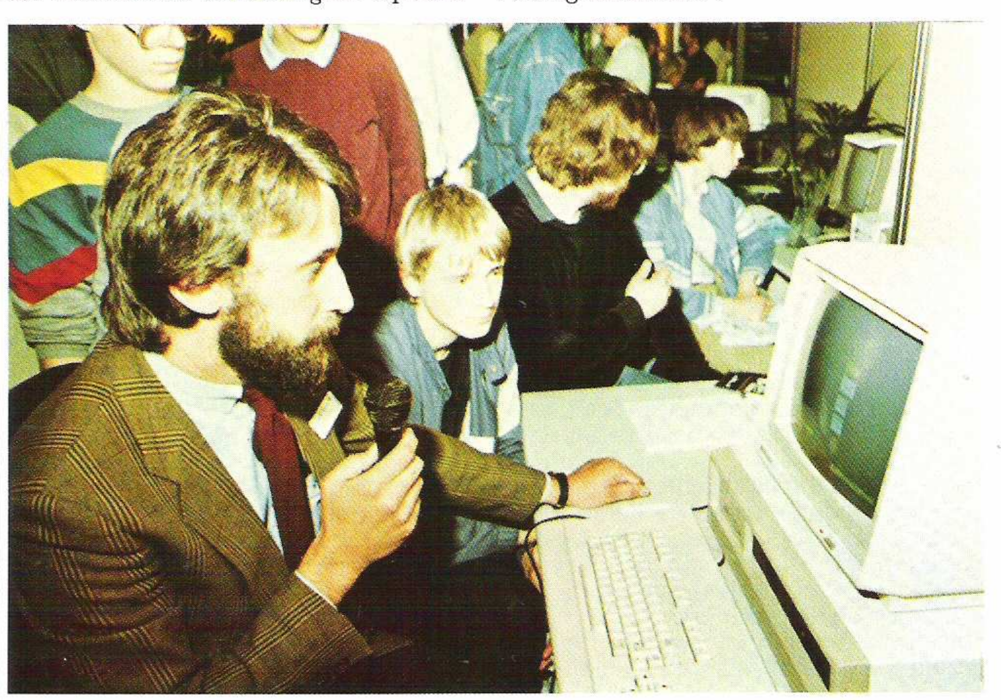
Schon geht der Dialog mit dem Computer weiter. „Wie alt bist Du, Jens?" fragt dieser auf dem Bildschirm, und nach dem drücken der Taste „4" die Antwort: „Du bist 4 Jahre alt. Danke, Jens!" Die älteren Kinder oder die Kindergärtnerinnen stehen hinter den kleinen Steppkes und lesen ihnen vor, wenn sie es nicht selbst können.

„Logo lernen mit Logo", heißt das nach einem Buch des amerikanischen Informatikers und Psychologen Pappert von einer Arbeitsgruppe am Lehrstuhl des Saarbrücker Erziehungswissenschaftlers Prof. Kötter entwickelte Computerprogramm. Eines seiner Grundprinzipien: Der Computer malt einen Igel auf den Bildschirm und mit Hilfe dieses Igels sowie Befehlen wie „vw" (vorwärts), „rw" (rückwärts) oder „re" (rechts) und „li" (links) lernen die Kinder, ein Haus samt Dach oder Kreise zu malen. Wichtig dabei ist jedoch auch, daß sie ihren „Bildschirm-Igel" die richtige Zahl von Schritten vorwärts, rückwärts oder nach rechts und links bewegen. „Auf solche Weise finden die Kinder einen ersten Bezug zur Mathematik", sagt der an der Projektleitung beteiligte Erziehungswissenschaftler Niegemann von der Saar-Universität.

Nicht computersüchtig

Die Skepsis der saarländischen Wissenschaftler und Kindergärtnerinnen in Ottweiler wegen des Modellversuchs „Kinder lernen am Computer" ist inzwischen vorsichtigem Optimismus gewichen. Nach anfänglich großer Begeisterung, so zeigte sich, lernen die meisten Kinder schon nach kurzer Zeit nur noch ein bis drei Mal pro Woche – und das in der Regel höchstens eine halbe Stunde hintereinander – am Computer.

Dann – so die Antworten einer Fragebogenaktion unter den 4- bis 14jährigen – „verliert man die Lust", „wird es langweilig" oder es „hängt einem zum Halse raus" – und man spielt lieber im Freien mit den anderen Fußball, turnt, läuft Rollschuhe oder übt Fahrradfahren. Andererseits wird der Computer von den Kindern als zusätzliche Spielmöglichkeit begeistert aufgenommen und vier von fünf der Kleinen glauben, daß sie dabei etwas lernen. „Schreiben und Lesen" beispielsweise – oder: „Man kann rechnen und der Computer sagt, ob es richtig oder falsch ist". Fazit der an dem saarländischen Modellversuch beteiligten Wissenschaftler und Pädagogen: Da die neue Technologie nicht aufzuhalten ist, sollte man durchaus schon im Kindesalter „versuchen, die Chancen, die der Computer bieten kann, zu nutzen, (aber auch) eventuelle Gefahren frühzeitig erkennen".



Warum also nicht schon 4- oder 6jährige mit dem Computer konfrontieren? Der kleine Jens möchte jedenfalls trotz oder ob seiner Computererfahrung „am liebsten Feuerwehrmann" werden.

Genau in diese Marktlücke hat sich die Firma ›Tobby – Disk International‹ mit ihrem Softwareangebot angepaßt. „Weg von der Gewalt", unter diesem Motto steht das Softwareangebot, wie: 1000 und eine Nacht, Puzzlespiele und Denkspiele. R.F.

Vorstellung: »Die Geschichte der Mikroelektronik«

Wo bitte liegt Silicon Valley?

Vierzig Meilen südlich von San Francisco entstand aus einer Ansammlung von Bauelemente- und Computerherstellern, Nobelpreisträgern, sonderlinghaften Wissenschaftlern, verbissenen Programmierern, verbohrten Technikern, Spezialisten für elektronische Kriegsführung und schwerreichen Kapitalanlegern das Nervenzentrum der mikroelektronischen Revolution: Silicon Valley.

Dirk Hanson, der sich in seiner journalistischen Arbeit auf die Wechselwirkung zwischen Mikroelektronik und Wirtschaft spezialisiert hat, recherchierte zwei Jahre lang in Silicon Valley. Sein Buch DIE GESCHICHTE DER MIKROELEKTRONIK vermittelt fundiertes Wissen, spannend erzählt. Gesehen bei: Heyne Computer Bücher, Heyne-Buch Nr. 15/3, DM 12,80. Deutsche Erstausgabe.

Forth-Kurs

Teil 6

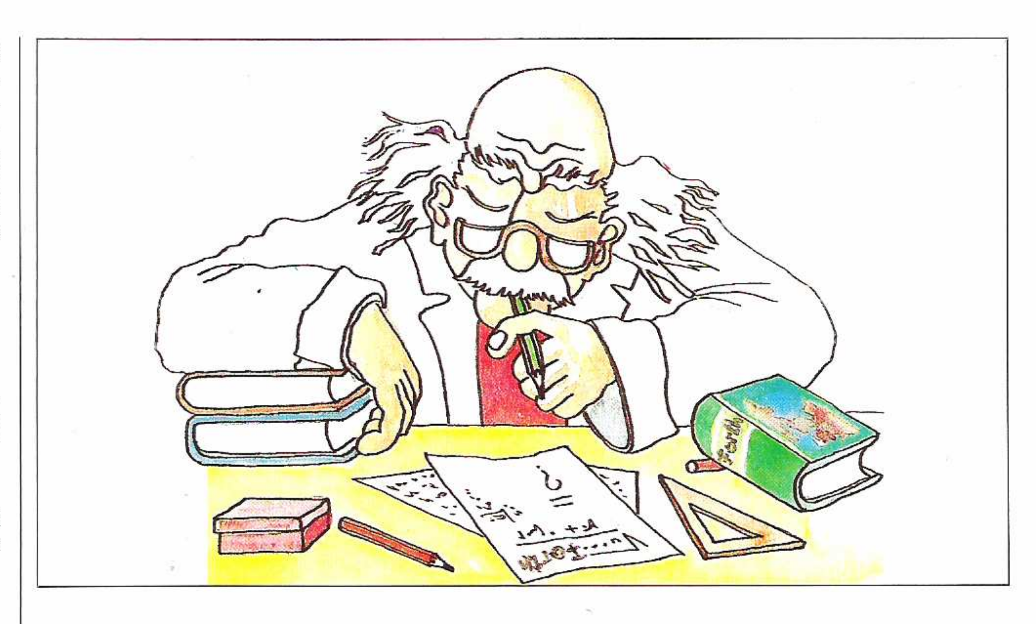
Die Endlosschleife Begin ... Again

Diese Programmschleife kennt keine Aussprungbedingung, ist also endlos. Die Hauptanwendung dieses Befehlswortes liegt im Aufruf bzw. in der unbegrenzt häufigen Abarbeitung eines (periodisch zu durchlaufenden) Hauptprogrammes. Die Syntax dieser Schleife entspricht weitgehend der BEGIN-UNTIL-Schleife.

Beispiel:

```
: TEST
        BEGIN
          . "COMPUTRONIC" CR (die-
se Anweisung wird endlos ausgeführt)
        AGAIN
;
```

Zu bemerken ist noch, daß es sich bei dieser Schleifenart um die schnellste FORTH-Schleife handelt.



BEGIN-WHILE-Schleife

Auch diese Schleifenstruktur wird beliebig lange durchlaufen. Im Gegensatz zur BEGIN-UNTIL-Schleife erfolgt die Abfrage einer Aussprungbedingung (Flag) am Anfang des Schleifenkernes. Ist die Bedingung am Schleifenbegin nicht erfüllt, so wird die Schleife nicht ein einziges Mal durchlaufen. Das folgende Beispiel zeigt die Syntax dieser Schleife:

```
BEGIN
...         (Bedingung zur Erzeugung
            des Flags)
WHILE       (solange das Flag = 1 ist,
...         wird der folgende Teil
            durchlaufen)
REPEAT      (Rücksprung nach BEGIN)
```

In dem folgenden Beispiel wird die Tastatur abgefragt. Ist die Taste „C" gedrückt, so wird der Text „TASTE C IST GEDRÜCKT" ausgegeben. Die Schleife wird erst verlassen wenn eine andere Taste gedrückt wird.

```
: TASTE
    BEGIN
      KEY 67 - 0 = (Ist Taste „C"
gedrückt?)
        WHILE
          . "TASTE C IST GEDRÜCKT"
CR
      REPEAT (Springe zurück)
;
```

Wir haben nun alle verfügbaren FORTH-Schleifen kennengelernt. Es gibt in FORTH auch die Möglichkeit noch weitere Arten von Schleifen zu definieren, doch dies ist dem etwas erfahreren Programmierern vorbehalten. Wer sich genauer damit beschäftigen möchte, muß auf andere Literatur zurückgreifen, da die Beschreibung dieser Definitionen den Rahmen dieses Kurses sprengen würde.

FORTH-Programmbeispiele:

Berechnung des größten gemeinsamen Teilers zweier Zahlen:

```
: TEILER (Größter gemeinsamer Teiler
errechnen)
    BEGIN
      SWAP OVER MOD DUP
      0 =
    UNTIL
    DROP
;
27 21 TEILER ‹RETURN› 3 OK
 8  6 TEILER ‹RETURN› 2 OK
```

Die Berechnung wurde mit Hilfe des Euklidschen Algorithmus verwendet. Zuerst bestimmt man den Rest ›R‹ zweier Zahlen, ›A‹ und ›B‹, durch (A MOD B). Ist R=0, dann ist B der gesuchte Teiler. Ist R‹›0, so wird A=B und B=R. Danach wir die obige Division wieder ausgeführt. Solange, bis die Bedingung R=0 erfüllt ist.

Variablen und Konstanten in Forth

Obwohl die meisten Zahlenwerte und Rechenoperationen über den Stack geführt werden, kann man trotzdem auch Konstante und Varibalen definieren.

In Basic wird eine Konstante wie folgt definiert:

```
10 LET Y = 10
```

Der Konstanten Y wird der Zahlenwert 10 zugeordnet. In FORTH würde das gleiche so geschrieben:

```
10 CONSTANT Y ‹RETURN› OK
```

Durch diese Befehlsfolge wird im Wörterbuch eine Konstante „Y" eingetragen, die den Zahlenwert 10 enthält. Sobald der Variablennamen aufgerufen wird, erhält der Stack den entsprechenden Zahlenwert.

Beispiele:

```
12 Y + .  ‹RETURN›   22 OK
 5 Y ★ .  ‹RETURN›   50 OK
 Y Y ★ .  ‹RETRUN›  100 OK
```

Das ganze kann auch in einer Definition stehen:

```
: Test
    Y ★ .
;

7 TEST ‹RETURN› 70 OK
```

Wie wir wissen, können wir statt der Konstanten auch einfach folgendes schreiben:

```
: Y 10 ;
```

Dies hat zwar die gleiche Wirkung wie die Definition einer Konstanten, ist jedoch in der Ausführungszeit wesentlich länger.

Im Gegensatz zu den meisten BASIC-Versionen können die Namen der Variablen, Konstanten und Befehlen in den meisten FORTH-Versionen bis zu 31 Zeichen lang sein. Es ist so möglich, den Variablen einen sinnvollen Namen zu geben, um somit das Programm übersichtlicher zu machen.

Ähnlich wie die Konstanten werden in FORTH auch die Variablen definiert:

10 VARIABLE TAG

In diesem Beispiel wurde der Variablen „TAG" der Wert 10 zugewiesen. Der Unterschied zwischen Variablen und Konstanten besteht darin, daß die Konstante nach der Definition ihren Wert immer behält. Dieser wird beim Aufruf der Konstanten auf den Stack gelegt. „Beim Aufruf einer Variablen wird dagegen nicht der Wert der Variablen, sondern die Adresse dieser Variable auf den Stack gelegt. Zum Ändern bzw. Auslesen des Wertes einer Variablen, benötigen wir noch zwei weitere Befehle, nämlich den Befehl :Wert holen (Load) ; und den Befehl :Wert wegspeichern (STORE).

Für den STORE-Befehl wird in Forth das „!" Zeichen verwendet. Dieser Befehl entspricht weitgehend dem bekannten POKE aus dem BASIC. Jedoch wird hier keine 8-Bit-Zahl, sondern eine 16-Bit-Zahl abgespeichert. Die Syntax des STORE-Befehl:

! (n1 n2 adr --› n1)

Für den LOAD-Befehl wird in Forth das " " Zeichen verwendet. Dieser Befehl ist mit dem Peek-Befehl im Basic zu vergleichen. Auch hier wird eine 16-Bit-Zahl ausgelesen:

Die Syntax des LOAD-Befehl:

@(n1 n2 adr --› n1)

Die eben vorgestellten Befehle LOAD und STORE arbeiten nur mit 16-Bit-Zahlen. Will man dagegen nur ein einziges Byte, also eine Dezimalzahl zwischen 0 und 255 abspeichern, so verwendet man das Wort C! „Mit Ce wird ein Byte auf den Stapel geholt. Dabei werden die oberen 8 Bits zu Null gesetzt.

Wenn wir nur den Variablennamen, gefolgt von einem Dezimalpunkt (Print) eingeben, so erhalten wir die Adresse der Variablen. Diese kann je nach FORTH-Version und Anzahl der Definitionen verschieden sein.

TAG . ‹RETURN› 4104 OK

Möchte man den Wert der Variablen TAG auslesen, so geschied dies wie folgt:

TAG @ . ‹RETURN› 10 OK

Das Ändern einer Variablen geht ebenso einfach:

13 TAG ! ‹RETURN› OK
(Der Wert 13 wird gespeichert)

TAG @ . ‹RETURN› 13 OK
(Der Wert wird ausgelesen)

Auf ähnliche Weise kann man auch den Wert einer schon vereinbarten Konstanten ändern. Die Adresse einer Konstanten erhält man mit dem Befehlswort "'" (Einzelapostroph).
Die Syntax:
' (Name) (n1 n2 --› n1 n2 adr)
Die Adresse der Konstanten (Name) wird auf den Stack gebracht.

Beispiele:

20 CONSTANT TRONIC ‹RETURN› OK
(der Konstanten TRONIC wird der Wert 20 zugewiesen)

TRONIC . ‹RETURN› 20 OK
(der Wert wird ausgelesen)

30 ' TRONIC ! ‹RETURN› OK
(der Wert 30 wird in die Konstante geschrieben)

TRONIC . ‹RETURN› 30 OK
(nochmal überprüfen)

FORTH-Zahlensystem

Bei der Initialisierung von Forth, wird immer das Rechnen zur Basis 10, also die gewohnte Dezimalrechnung angenommen. Forth kann jedoch auch mit allen anderen Zahlensystemen rechnen. Will man z. B. mit Hexadezimalzahlen arbeiten, so kann mit dem Befehl „HEX" auf diesen Modus umgeschaltet werden. Danach erfolgen alle Ein- und Ausgaben in hexadezimaler Schreibweise.

HEX (n1 n2 n3 --› n1 n2 n3)

Möchte man wieder in den Dezimal-Modus zurück, so geschieht dies mit dem Befehl "DECIMAL".

DECIMAL (n1 n2 n3 --› n1 n2 n3)

Beispiel:

DECIMAL 90 HEX . ‹RETURN› 5A OK
Die Dezimalzahl 90 entspricht der Hexadezimalzahl 5A.

HEX 5A DECIMAL . ‹RETURN› 100 OK
Das umgekehrte Beispiel.

Möchte man ein anderes Zahlensystem verwenden, muß man in die System-Variable BASE den entsprechenden Wert speichern. Durch den Befehl HEX wird eine 16 und durch DECIMAL eine 10 in die Variable geschrieben.

Möchten wir beispielsweise das Binärsystem programmieren, so müssen wir nur die Zahl 2 (Zustand 0 und 1) in die Variable BASE bringen. Den Umschalt-Befehl nennen wir BIN.

: BIN 2 BASE ! ;

Mit dem Befehl BIN können wir nun jederzeit auf Binär-Modus umschalten.

```
DECIMAL 255 BIN .
      ‹RETURN› 11111111 OK
DECIMAL 16 BIN .
      ‹RETURN›    10000 OK
BIN 01010101 DECIMAL .
      ‹RETURN›       85 OK
BIN 00010000 DECIMAL .
      ‹RETURN›       16 OK
```

Zum Abschluß noch ein kleines FORTH-Programm:

Berechnung der Primzahlen zwischen zwei Grenzen.

```
4 VARIABLE ZEILE
: TEST      MOD 0=;
: DRUCKE DUP 4 .R ZEILE e DUP 0=
            IF CR DROP 4
            ELSE 1 -
            THEN
            DROP ;

: PTEST     DUP 2 / 2 DO DUP I TEST
            IF 0 LEAVE
            THEN  LOOP  DUP  IF
            DRUCKE
            ELSE DROP
            THEN
            DROP ;

: PRIM      (ende anf --› )
CR 4        ZEILE !
DO I        PTEST LOOP CR ;

200 1 PRIM ‹RETURN›

  1    2    5    7   11
 13   17   19   23   29
 31   37   41   43   47
 53   59   61   67   71
 73   79   83   89   97
101  103  107  109  113
127  131  137  139  149
151  157  163  167  173
179  181  191  193  197
199  OK
```

Für unsere Atari-Freunde

1. Tip:
Der einfachste Weg, sich eine komplette Variablenliste zu verschaffen ist folgender: Laden Sie das Programm (oder geben Sie es ein). Schreiben Sie dann SAVE"S:" und drücken Sie RETURN. Drücken Sie BREAK wenn etwa die Hälfte des Bildschirms mit Zeichen gefüllt ist. Das 15. Zeichen von links oben ist das erste Zeichen der Variablenliste, die mit (mindestens) einem Herzchen (CHR$(O)) endet. Der letzte Buchstabe eines Namens ist immer invers dargestellt. Besteht der Name aus nur einem Buchstaben, so ist dieser invers.
Beispiele:

NAME	(_=invers)
A	(Variable a)
NAME$	(Sring-Variable)
FELD(	(DIMensioniertes Feld)
CDE	(Die Variablen C, D, E hintereinander)

Auf dem Bildschirm sieht das so aus:
NAMEANAME$FELD(CDE♥...
Die Variablen sind in der Reihenfolge ihrer Eingabe geordnet. Auf dem Bildschirm sieht man genau das, was durch Befehle SAVE"C:" bzw. SAVE"D:(name)" auf Cassette bzw. Disk geschrieben würde. Mit SAVE"P:" (P=Printer, S=Screen) könnte man das ganze auch auf dem Drucker ausgeben.
Oft geistern in dieser Liste Namen herum, die im Programm gar nicht mehr benutzt werden. Wenn man sich z. B. bei der Eingabe eines Namens vertippt hat und den Fehler erst nach dem Drücken von RETURN bemerkt, dann hat man einen neuen Namen in der Liste, der sich auch mit CLR (alle Variablen löschen) nicht mehr beseitigen läßt.
Wenn Sie das Programm jedoch mit LIST"C:" bzw. LIST"D:(name)" abspeichern und danach mit „ENTER"... wieder laden, dann wird eine neue Liste erstellt, die wirklich nur die im Programm vorkommenden Namen enthält.

2. Tip:
Den Ausdruck eines Fragezeichens bei der Ausführung des INPUT-Befehls kann man mit folgender Prozedur verhindern:

OPEN #X,4,0,"E:"	(eröffnet einen Datenkanal vom EDITOR)
INPUT #X,NAME	(X=1-7, jedoch nicht 6)
CLOSE #X	

Wenn mehrere INPUTs gemacht werden müssen, dann folgt der CLOSE-Befehl (Datenkanal schließen) nicht sofort.
NAME kann natürlich auch eine String-Variable sein.
GRAPHICS 0 ist das gleiche wie OPEN #6,12,0,"S:", daher kann man für X keine 6 einsetzen, wenn vorher ein Graphikmodus eröffnet wurde.
Wenn sie schreiben OPEN #X,9,0,"E:", dann können Sie mit dem INPUT-Befehl (wie oben) etwas einlesen, ohne daß der Benutzer RETURN gedrückt hat.

Beispiel:
Dieses Programm liest ein Wort in die Variable A$ ein, sobald die horizontale Position des Cursors größer als 10 ist.

```
10 DIM A$(10)
20 GRAPHICS 0: REM ENTSPRICHT DATENKANAL VOM/ZUM BILDSCHIRM
30 OPEN #1,9,0,"E:": REM DATENKANAL VOM EDITOR
40 OPEN #2,4,0,"K:": REM DATENKANAL VOM KEYBOARD
50 GET #2,T: REM CODE EINER GEDRÜCKTEN TASTE HOLEN
60 PRINT CHR$(T);: REM ZEICHEN AUSDRUCKEN
70 IF PEEK (85)≦10 THEN 50: REM HORIZONTALE CURSORPOSITION
80 POSITION 2,0: REM CURSOR AN DIE ANFANGSPOSITION
90 INPUT #1,A$
100 PRINT "A$="; CHR$(34); A$; CHR$(34): REM ERGEBNIS
110 END:REM IMPLIZIERT _CLOSE #1:CLOSE #2_
```

Mit dem Befehlt END kann man alle Datenkanäle auf einmal schließen. CHR$(34) ergibt ein Ausführungszeichen.
Unabhängig davon, was der Benutzer eingibt, es werden nur die Zeichen 2–10 in der Zeile 0 gelesen, sobald die horizontale Position des Cursors größer als 10 ist.

3. Tip:
Programmteile per Programm löschen.

Probieren Sie folgendes:

```
10 GRAPHICS 0
20 POSITION 2,4
30 ? 80: ? 90: ? 100: ? "CONT"
40 POSITION 2,0
50 POKE 842,13
60 STOP
70 POKE 842,12
80 REM DIESE ZEILEN
90 REM WERDEN
100 REM GELÖSCHT
```

Durch POKE 842,13 wird der Computer veranlaßt, wild drauflos zu lesen und auszuführen was er liest, sobald er die Programmausführung beendet hat.
Durch die Positionierung in Zeile 40 steht der Cursor nach Ausdruck einer STOPPED AT LINE-Meldung auf der Zahl 80.
Er liest sie, als ob der Benutzer sie eingegeben und RETURN gedrückt hätte. Führt den "Befehl" aus und löscht Zeile 80.
So geht das weiter, bis er etwas liest, das ihn veranlaßt im Programm weiterzumachen, also z. B. CONT (=CONTINUE), RUN, GOTO 70 oder ganz einfach POKE 842,12.
Die Speicherstelle 842 wird im OPERATING SYSTEM SOURCE LISTING übrigens ICAX1 (AUXILLARY INFORMATION) genannt.
Mit dieser Methode kann man natürlich auch ganze Zeilen per Programm ins Programmm holen.

4. Tip:
Wenn man ein Programm mit LISTI"C:" bzw. LIST"D:(name)" auf Cassette bzw. Disk abspeichert, dann wird es dorthin als Klartext geschrieben, d. h. man kann z. B. einen Programmtext von Cassette oder Disk laden und bearbeiten.

Beispiel:
Ein Programm wird mit LIST"D:TEST" auf Diskette geschrieben. Das folgende Programm liest die einzelnen Zeilen und druckt sie als Text aus.

```
10 OPEN #1,4,0,"D:TEST": REM DATENKANAL
20 DIM A$(128)
30 TRAP 70: REM BEI ERROR-136 SPRUNG NACH 70
40 INPUT #1, A$: REM ZEILE LESEN
50 PRINT A$: REM UND AUSDRUKKEN
60 GOTO 40: REM NÄCHSTE ZEILE
70 PRINT
80 PRINT "ERROR-END OF FILE"
90 END: REM DATENKANAL SCHLIESSEN
```

Anstatt die Zeilen nur auszudrucken kann man sie natürlich auch bearbeiten und in eine zweite Datei schreiben, um sie von dort mit dem Befehl ENTER "D:(name)" als neues Programm zu laden. Syntax-Fehler werden dabei angezeigt.

Beispiel:
Dieses Programm liest die Zeilen eines mit LIST"D:TEST" abgespeicherten Programms und legt sie in der Datei "D:TEST.NEU" wieder ab. Zeilen, die nur aus einer REM-Anweisung bestehen, werden jedoch nicht übertragen (siehe Zeile 80).

```
10 DIM A$ (128)
20 OPEN #1,4,0,"D:TEST"
30 OPEN #2,8,0,"D:TEST.NEU"
40 TRAP 110
50 INPUT #1,A$
60 Q=VAL (A$): REM Q=ZEILENNUMMER
70 L=LEN(STR$(Q))+2: REM L=LÄNGE DER ZEILENNUMMER +2
80 IF A$(L,L+3)="REM" THEN 50
90 PRINT #2;A$
100 GOTO 50
```

110 PRINT "FERTIG!"
120 END
Das neue Programm kann nun mit ENTER "D:TEST.NEU" geladen werden.
ACHTUNG: in Zeile 80 muß hinter REM ein (!) Leerzeichen stehen, sonst würde z. B. die Zeile REMEMBER=5:... gelöscht! Man kann auch ganz neue Zeilen (z. B. DATA-Zeilen) schreiben lassen.

Tips&Tricks für den DRAGON-USER

Obwohl das Informationsblatt für MC-USER von DRAGON schon viele Hinweise auf ROM-Routinen und ZERO-PAGE-Adressen enthält, lassen sich darüber hinaus noch viele sinnvolle Routinen finden.

So dürfte wohl den meisten DRAGON-USER der Befehl: EXEC &H8C9C unbekannt sein. Diese ROM-Routine entfernt scheinbar die Zeilennummern aus dem Programm. Dadurch ist es möglich, Texte ohne REM-Zeichen in Zeilen aufzulisten und anschließend als zusammenhängenden Text auf seinem Drucker auszugeben.

Auch nach einem normalen LIST-Befehl erscheint das Programm ohne Zeilennummern. Will der USER wieder Ordnung durch Nummern haben, braucht er lediglich „RUN" einzugeben und schon ist sein altes Programm wieder richtig hergestellt.

Die DRAGON-USER, die eigene Programme schreiben und auch auf Cassette weitergeben wollen, machen sich oft Gedanken über einen entsprechenden Programmschutz. Es sollte jedoch klar sein, daß es einen solchen Schutz für wirklich alle Fälle kaum gibt, denn die Tricks zum „Knacken" eines Programmes sind schier unerschöpflich.
Dennoch gibt es einige Tricks, die die „Knacker" schon zum Überlegen anregen. Dazu gehört u. a., daß das geschützte Programm nicht mehr gelistet werden kann. Um dies zu erreichen, muß immer über „CSAVEM" abgespeichert werden, um schon während dem Laden die entsprechenden Speicherstellen mit Anweisungen zu versehen.
Da jedes BASIC-Programm mit „CSAVEM" abgespeichert werden kann, muß nur der Speicherbereich richtig angegeben werden. Dazu muß über die Speicherstellen 25, 26, 27 und 28 der jeweilige BASIC-Speicherbereich erfragt werden.
Ich habe eine ROUTINE entwickelt, mit deren Hilfe man seine Programme relativ einfach vor „Listen" schützen kann.
Diese ROUTINE setzt sich aus mehreren Schutzbefehlen zusammen:
Ausschalten der BREAK-Taste
Ausschalten der RESET-Taste
Ausschalten der Unterbrechung nach einem Fehler
Eingabekontrolle unmittelbar nach dem Laden

Geschaltet wird das MC-Programm über die Speicherstelle &H167. Diese wird durch Eingabe eines Zeichens vom ROM aus angesprungen und enthält normal ein RTS (&H39).
Die beiden auf diese Speicherstelle folgenden Bytes ergeben nach hexadezimaler Zusammensetzung eine Sprungadresse, wenn statt RTS in &H167 ein JMP-Befehl (&H7E) steht.
Da diese Stelle unmittelbar nach dem Laden angesprungen wird, ist es möglich, durch ein entsprechendes MC-Programm sein BASIC-Programm schon vor einem „RUN"-Befehl, also während des Einladens vor „LIST" zu schützen.
Ich habe die folgende Routine in die erste Grafikseite gesetzt. Für USER, die diesen Bereich für gleichzeitiges Einladen von Grafikteilen auf direkte Art benutzen müssen, muß die Routine an einen anderen Bereich verschoben werden. Dann müssen auch die Sprunganweisungen entsprechend angepaßt werden.

Hier ist das ASSEMBLERLISTING der Schutzroutine

```
DISSABLE LIST
1536 600 3402      PSHS A.
1538 602 814C      CMPA #4C
1540 604 27FC      BEQ $602
1542 606 8150      CMPA #50
1544 608 27F8      BEQ $602
1546 60A 8141      CMPA #41
1548 60C 27F4      BEQ $602
1550 60E 8145      CMPA #45
1552 610 27F0      BEQ $602
1554 612 3502      PULS A.
1556 614 39        RTS
1557 615 12        NOP
1558 616 8E0167    LDX #0167
1561 619 86BD      LDA #BD
1563 61B A784      STA ,X
1565 61D 7E8006    JMP $8006
```

Um dieses Programm nutzen zu können, müssen zunächst die Speicherstellen für die Eingabe eines Zeichens geändert werden.
Die Befehle dazu lauten:
POKE &H169, 0:POKE &H168, &H06:POKE &H167, &HBD
Durch diese POKEs wird das Maschinenprogramm angesprungen und schaltet LIST-LIST-SAVE-EDIT-DEL-POKE-PEEK usw. aus. Nur die Eingabe „RUN" als Befehl ist noch zulässig.
Nach Eingabe des Befehls „RUN" muß der Sprungbefehl zur MC-Routine unbedingt wieder aufgehoben werden.
Dazu ist der POKE &H167, &H39 erforderlich.
Weiterhin muß unmittelbar nach Programmstart der Befehl „LIST" durch POKE &H120, 19 ausgeschaltet werden.
Da es immer noch möglich ist, das Programm durch die BREAK-Taste oder die RESET-Taste zu stoppen, müssen auch diese Tasten ausgeschaltet werden.
Zuerst wird die MC-Routine in den Zeilen 10 bis 40 eingegeben. Es ist ratsam, sich diesen Teil zunächst für spätere Fälle abzuspeichern. Nach „RUN" ist der MC im Speicher

```
10 FORI = 0TO31: READP$: POKE &H0600+I, VAL ("&H"+P$): NEXTI
20 DATA 34, 02, 81, 4C, 27, FC, 81, 50, 27, F8, 81, 41, 27, F4, 81, 45, 27, F0, 35, 02, 39, 12, 8E, 01, 67, 86, 51, A7, 84, 7E, 80, 06
30 POKE&H169, &H0: POKE&H168, &H06
40 POKE&H71, &H55: POKE&H72, &H06: POKE&H73, &H15
50 NEW
```

Jetzt kann das BASIC-Programm geladen werden, das dann geschützt abgespeichert wird. Dazu müssen einige Programmergänzungen eingebaut werden

```
10 POKE27, 31: POKE28, 221: POKE&H167, &HBD: POKE&H120, 19: CLS0
20 POKE&H19B, &HE4: POKE&H19C, &HCB: POKE&H19D, &H4: POKE&H19E, &HED: POKE&H19F, &HE4: POKE&H19A, &HEC: POKE&H73, &H15: POKE&H72, &H0: POKE&H71, &H5530 A=0
40 I$=INKEY$: A=A+1
50 IFI$="X"THEN80
60 IFA<150THENI$="": GOTO40ELSEPOKE&H167, &H39: GOTO100
70 GOTO40
80 A=0: CLS: POKE&H167, &H39: PRINT§99, "PROGRAMMSCHUTZ";: POKE&H167, &HBD: CSAVEM"SCHUTZ", &H167, 9000, &H167: POKE&H167, &H39
90 '----ab Zeile 100 beginnt das BASIC-Programm----
100 CLS5
110 PRINT§RND(480), "SCHUTZSIMULATION";: SOUNDRND(255), 1
120 CLSRND(9)-1: SOUNDRND(255), 1: GOTO110
```

Die Zeilen 10 bis 100 müssen in dieser Reihenfolge in das zu schützende Programm eingebaut werden. Ab Zeile 100 beginnt dann das normale BASIC-Programm.
Der Schutz kann noch beliebig erweitert werden, wie z. B. durch ERROR-Routinen oder „RUN" nach RESET usw.
Mit dieser kurzen Simulation dürfte vielen USERN der Einstieg in eigene Schutzentwicklungen erleichtert werden.

(K.-H. Waldscheck)



REVERSI

Dieses Programm ist absichtlich nur teilweise in M.C.-Code geschrieben. Um eigene Vorstellungen des Spielbrettes oder Anfangspositionen zu verwirklichen, ist BASIC nun einmal besser geeignet. Außerdem läßt sich das M.C.-Programm relativ einfach für andere Computer mit Z80 CPU umschreiben.

Das BASIC-Programm erklärt sich durch die REM-Zeilen zum Teil selbst. Darum nur eine Beschreibung des M.C.-Programmes.

Um vom BASIC-Programm die Spielfelddaten ohne große Umrechnungen aus dem M.C.-Programm abrufen zu können, ist ein geschlossener 100er-Block (16600–16700) Speicherzellen reserviert worden. Es werden für die Daten zwar nur 64 Speicherzellen benötigt, aber der Computer möchte gerne wissen, wo der Spielfeldrand beginnt. Diese Speicherzellen werden bei jedem Spielbeginn vom BASIC-Programm initialisiert und können bei der Eingabe des M.C.-Programmes übersprungen werden. In den Zeilen 16515, 16516 steht die Position des Zuges, welcher a) der Spieler zu setzen wünscht (wird von BASIC-Zeile 502–517 gepokt) oder b) vom Computer hinterlegt (um in Zeile 720–732 seinen Zug bekannt zu geben).

Ab 16520 beginnt das Hauptprogramm für den Spielerzug. Hier wird im Kreis um die gesetzte Position abgefragt, ob ein gegnerischer Stein geschlagen werden kann. Wenn dies der Fall ist, werden die gegnerischen Steine umgedreht (dieser Programmteil wird auch vom Computer als erstes Unterprogramm benutzt). Danach werden im separat aufgerufenen M.C.-Programm USR 17495 die Steine gezählt. Hat sich an der Anzahl der Steine nichts gändert muß der Zug falsch gewesen sein und der Computer meldet dies. Durch betätigen von N.L. darf man den Zug wiederholen. War der Zug gültig, wird in das Unterprogramm (ab Zeile 100) gesprungen und die Spielerfiguren neu gesetzt.

Das Programm des Computers ist wesentlich umfangreicher. Es beginnt bei Speicherzelle 17020. Im wesentlichen sind in diesem Programmteil die Prioritäten festgelegt (die für den Spielverlauf wichtigsten Positionen werden zuerst abgefragt, in diesem Fall A1, H8, H1, A8). Hat der Computer einen freien Platz gefunden, springt er nach 16520 und fragt die Positionen ringsum ab. Anschließend springt er in das Unterprogramm 16940 und zählt seine Steine. Hat sich der Zählerstand nicht geändert, springt er zurück und bearbeitet die nächste Priorität. Hat sich der Zählerstand jedoch geändert, springt er in einen Programmteil (16961), welcher auf den ersten Blick einen Programmfehler vermuten läßt. Hier wird 2mal ge„popt", obwohl vorher nicht ge„pusht" worden ist. Normalerweise führt dies zum Absturz des Rechners. Hier hebt er jedoch 2 CALL Rücksprungadressen ab, um die Rücksprungadresse nach BASIC zu finden.

Wie schon beschrieben, ist die Strategie des Computers in den Speicherzellen 17030–17494 festgelegt. Da der Computer nicht nach Quantität sondern nach Qualität seine Spielentscheidung trifft, aber dabei nicht prüft, ob an anderer Stelle ein gleichwertiger Zug mehr Steine schlagen kann, wäre es vorteilhaft, die Strategie von Zeit zu Zeit zu ändern.

Nach der Code-Nr. 33 (ld hl,NN) folgt eine Adresse des Spielfeldes. Werden diese getauscht, fällt der Computer auch andere Spielentscheidungen.

Obwohl der Computer keinen versierten Spieler schlagen kann, ist seine Strategie gar nicht so ohne und versetzt den ungeübten Spieler manchmal ganz schön ins Schwitzen.

```
 1 REM MIN.1030 ZEICHEN EING.
 2 REM DANACH MIT RUN STARTEN UND
 3 REM MASCHINENCODE EINGEBEN
 4 REM NACH DER EINGABE ALLER ZAHLEN
 5 REM DIE ZEILEN 2 BIS 15 LOESCHEN.
 6 REM NUN HAUPTPROGRAMM EINGEBEN .
 7 FOR X=16514 TO 17533
 8 SCROLL
 9 PRINT "SPEICHERTELLE ";X
10 SCROLL
11 INPUT Y
12 SCROLL
13 PRINT X,Y
14 POKE X,Y
15 NEXT X
```



M.C. LISTING REVERSI

16514	16534	16554	16574	16594
33	0	43	190	0
38	204	190	204	0
65	102	0	240	0
201	65	204	65	0
0	6	180	35	0
40	8	65	190	0
205	43	205	204	128
130	16	130	14	128
54	253	64	66	128
62	190	6	201	128
151	0	9	0	128
35	204	35	0	128
190	124	16	0	128
0	65	253	0	128
204	43	190	0	128
80	190	0	0	128
65	0	204	0	128
43	204	210	0	151
43	152	65	0	151
190	65	35	0	151

16700	16720	16740	16760	16780
28	229	225	252	190
28	6	201	121	32
28	2	229	225	4
28	35	6	201	119
28	4	2	229	25
28	190	43	17	16
28	40	4	9	252
28	251	190	0	121
28	79	40	6	225
28	58	251	2	201
28	135	79	237	0
28	64	58	82	0
28	190	135	4	229
28	32	64	190	17
28	4	190	40	10
28	119	32	250	0
28	43	4	79	6
28	16	119	58	2
28	252	35	135	237
28	121	16	64	82

16800	16820	16840	16860	16880
4	229	25	251	229
190	17	16	79	17
40	11	252	58	10
250	0	121	135	0
79	6	225	64	6
58	2	201	190	2
135	237	0	32	25
64	82	0	5	4
190	4	0	119	190
32	190	0	237	40
4	40	229	82	251
119	250	17	16	79
35	79	9	251	58
16	58	0	121	135
252	135	6	225	64
121	64	2	201	190
225	190	25	0	32
201	32	4	0	5
0	4	190	0	119
0	119	40	0	237

16900	16920	16940	16960	16980
82	251	1	200	0
16	79	0	113	0
251	58	78	33	0
121	135	62	135	0
225	64	40	64	0
201	190	33	54	0
0	32	227	151	0
0	5	64	33	0
0	119	190	140	0
0	237	32	64	0
229	82	1	54	0
17	16	12	40	0
11	251	35	209	0
0	121	16	209	0
6	225	249	201	0
2	201	33	0	0
25	31	40	0	0
4	33	66	0	0
190	0	121	0	0
40	0	150	0	0

17000	17020	17040	17060	17080
175	33	104	32	33
190	135	66	18	48
192	64	33	33	65
34	54	41	236	62
131	40	65	64	40
64	33	205	205	190
0	140	104	104	32
0	64	66	66	18
0	54	33	33	33
0	151	48	237	37
0	33	65	64	65
0	227	205	205	205
205	64	104	104	104
136	205	66	66	66
54	104	33	33	33
205	66	227	228	38
44	33	64	64	65
66	234	62	205	205
201	64	40	104	104
0	205	190	66	66

17100	17120	17140	17160	17180
33	33	33	64	104
47	244	32	205	66
55	64	65	104	33
205	205	205	66	21
104	104	104	33	65
66	66	66	46	205
33	33	33	65	104
234	233	31	205	66
64	64	65	104	33
62	205	205	66	254
40	104	104	33	64
190	66	66	232	205
32	33	33	64	104
18	41	42	205	66
33	65	65	104	33
243	62	205	66	28
64	40	104	33	65
205	190	66	43	205
104	32	33	65	104
66	18	229	205	66

17200	17220	17240	17260	17280
33	65	104	33	65
247	205	66	3	205
64	104	33	65	104
205	66	250	205	66
104	33	64	104	33
66	252	205	66	18
33	64	104	33	65
249	205	66	16	205
64	104	33	65	104
205	66	24	205	66
104	33	65	104	33
66	251	205	66	1
33	64	104	33	65
26	205	66	8	205
65	104	33	65	104
205	66	6	205	66
104	33	65	104	33
66	25	205	66	230
33	65	104	33	64
23	205	66	11	205

17300	17320	17340	17360	17380



104	33	64	104	33
66	248	205	66	239
33	64	104	33	64
45	205	66	7	205
65	104	33	65	104
205	66	2	205	66
104	33	65	104	33
66	27	205	66	36
33	65	104	33	65
231	205	66	242	205
64	104	33	64	104
205	66	17	205	66
104	33	65	104	33
66	22	205	66	240
33	65	104	33	64
44	205	66	33	205
65	104	33	65	104
205	66	12	205	66
104	33	65	104	33
66	253	205	66	35

17400	17420	17440	17460	17480
65	104	33	64	104
205	66	38	205	66
104	33	65	104	33
66	243	205	66	42
33	64	104	33	65
241	205	66	228	205
64	104	33	64	104
205	66	237	205	66
104	33	64	104	33
66	238	205	66	13
33	64	104	33	65
34	205	66	47	205
65	104	33	65	104
205	66	31	205	66
104	33	65	104	201
66	37	205	66	1
33	65	104	33	0
32	205	66	233	78
65	104	33	64	62
205	66	244	205	40

17500	17520	17540	17560	17580
33	227	28	138	0
227	64	28	122	118
64	190	28	0	0
190	32	118	0	4
32	1	0	0	11
1	12	1	118	0
12	35	2	0	237
35	16	0	3	29
16	249	234	12	28
249	33	118	0	28
33	41	0	237	126
40	66	2	30	135
66	113	12	28	72
113	201	0	28	0
1	28	237	28	0
0	28	29	126	0
78	28	28	139	118
62	28	28	122	0
151	28	28	0	5
33	28	126	0	11

EINGABE BIS 17533

R E V E R S I

VON

PAUL WEBRANITZ

█=SPIELERSTEIN C=COMPUTERSTEIN

EINGABE DER ZUEGE=BUCHST.,ZAHL

ZWISCHEN DEM ZU SETZENDEN STEIN UND DEM NAECHSTEN EIGENEN STEIN MUSS MINDESTENS EIN COMPUTER= STEIN LIEGEN

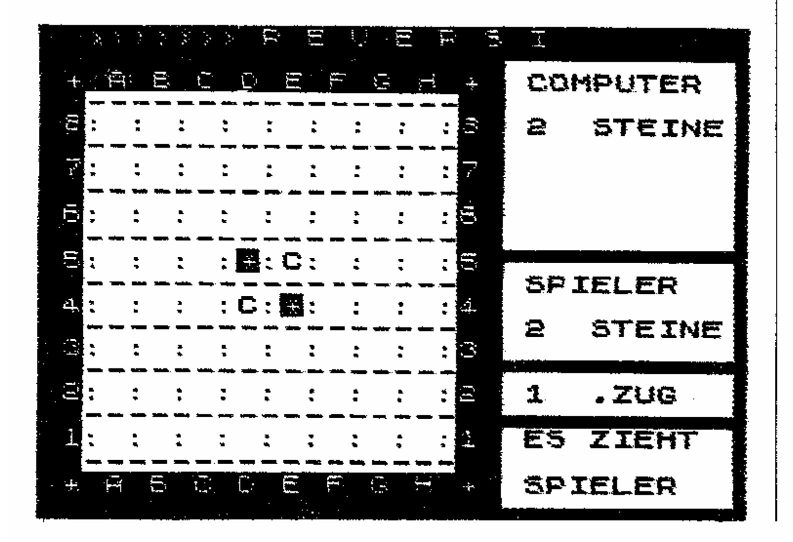
BEI EINGABE EINES FALSCHEN ZUGES N.L.DRUECKEN

SOLL DER COMPUTER ANFANGEN,"0" EINGEBEN

(AUCH WENN MAN SELBST NICHT MEHR ZIEHEN KANN,"0" EINGEBEN)

A L L E S   K L A R   ?

N.L. DRUECKEN



BASIC LISTING

```
  1█REM (MIN.1030 ZEICHEN)
  2 GOSUB 1000
  3 GOSUB 2000
  5 RAND USR 17495
  9 GOSUB 100
 10 GOTO 500
 99 REM STEINE SETZEN
100 LET N=16611
105 FOR X=3 TO 17 STEP 2
110 FOR Y=18 TO 4 STEP -2
115 PRINT AT Y,X;"*"
120 PRINT AT Y,X;CHR$ PEEK N
125 LET N=N+1
130 NEXT Y
135 LET N=N+2
140 NEXT X
150 PRINT AT 4,A;K$;AT 4,A;PEEK
L;AT 13,A;K$;AT 13,A;PEEK K
160 LET MC=PEEK L
161 LET SC=PEEK K
162 IF MC+SC=64 THEN GOTO 900
165 RETURN
499 REM SPIELERZUG
500 PRINT AT 20,A;S$
501 INPUT Z$
502 IF Z$="0" THEN GOTO 700
510 IF LEN Z$<>2 THEN GOTO 500
511 IF CODE Z$(1)<38 OR CODE Z$
(2)>45 THEN GOTO 500
515 LET P=16600+((CODE Z$(1)-37
)*10)+(CODE Z$(2)-28)
516 POKE 16516,INT (P/256)
517 POKE 16515,(P-INT (P/256)*2
56)
530 IF PEEK P<>CODE " " THEN GO
TO 800
540 RAND USR 16520
541 RAND USR 17495
```




```
 542 IF SC=PEEK K THEN GOTO 800
 550 GOSUB 100
 560 GOTO 700
 699 REM COMPUTERZUG
 700 PRINT AT 20,A;C$
 702 RAND USR 17020
 704 RAND USR 17495
 710 IF MC=PEEK L THEN GOTO 900
 715 LET B=(PEEK 16515+256*PEEK
16516)-16600
 720 LET U$=STR$ B
 724 FOR N=1 TO 10
 725 PRINT AT 6,21;" ICH SETZE "
;AT 6,21;" ICH SETZE ";AT 8,25;"
  ";AT 8,25;CHR$ ((VAL U$(1)+37)
);U$(2)
 730 FOR M=1 TO 5
 731 NEXT M
 732 NEXT N
 740 GOSUB 100
 745 LET Z=Z+1
 750 PRINT AT 16,A;Z
 760 GOTO 500
 799 REM FEHLERMELDUNG
 800 PRINT AT 20,A;" FALSCH "
 801 PRINT AT 20,A;" FALSCH "
 802 IF INKEY$="" THEN GOTO 800
 803 GOTO 500
 899 REM SIEGERERMITTLUNG
 900 IF MC=0 OR MC<SC THEN PRINT
 AT 13,20;"   SIEGER   "
 901 IF MC>SC OR SC=0 THEN PRINT
 AT 6,20;"   SIEGER   "
 910 PRINT AT 0,0;"  NEUES SPIEL
 N.L. DRUECKEN   "
 920 INPUT N$
 930 RUN
 999 REM SPIELFELD "AUFBAEUMEN"
1000 FAST
1001 LET L=16936
1002 LET K=16937
1003 FOR N=16600 TO 16699
1004 POKE N,0
1006 NEXT N
1010 FOR N=16600 TO 16609
1012 POKE N,128
1014 POKE N+90,128
1016 NEXT N
1020 FOR N=16600 TO 16690 STEP 1
0
1022 POKE N,128
1024 POKE N+9,128
1026 NEXT N
1032 POKE K,2
1033 POKE L,2
1034 POKE 16519,151
1035 POKE 16524,40
1036 LET S=16645
1044 SLOW
1045 RETURN
1049 REM ANFANGSPOSITION SPIELS
TEINE
1050 POKE S,151
1051 POKE S+9,151
1052 POKE S-1,40
1053 POKE S+10,40
1055 RETURN
1060 GOSUB 1050
1061 POKE S-9,40
1062 POKE S-12,151
1063 POKE S+21,151
1064 POKE S+18,40
1065 RETURN
1070 GOSUB 1050
1071 POKE 16623,151
1072 POKE 16632,40
1073 POKE 16662,40
1074 POKE 16673,151
1075 POKE 16626,40
1076 POKE 16637,151
1077 POKE 16667,151
1078 POKE 16676,40
1079 RETURN
1080 GOSUB 1050
1081 POKE 16611,151
1082 POKE 16618,40
1083 POKE 16681,40
1084 POKE 16688,151
1090 RETURN
1999 REM SPIELFELD
2000 LET C$="COMPUTER"
2001 LET S$="SPIELER "
2002 LET Z=1
2005 LET K$="   STEINE"
2006 PRINT AT 8,0;"
"
2007 PRINT "    WELCHES SPIELFELD
 "
2008 PRINT "          1-4
 "
2009 PRINT "
 "
2013 LET Q$=INKEY$
2014 IF Q$="1" THEN GOSUB 1050
2015 IF Q$="2" THEN GOSUB 1060
2016 IF Q$="3" THEN GOSUB 1070
2017 IF Q$="4" THEN GOSUB 1080
2018 IF Q$="" THEN GOTO 2013
2019 LET T=157
2024 PRINT AT 0,0;"          R E
 V E R S I          "
2025 PRINT AT 1,0;"
 "
2030 PRINT " + A B C D E F G H +
 "
2035 PRINT AT 20,0;" + A B C D E
 F G H + "
2040 PRINT "
 "
2045 FOR A=3 TO 18 STEP 2
2050 PRINT AT A,0;"█-----------
-----█"
2055 PRINT "█: : : : : : : : :█
█"
2060 PRINT "█-----------------█
 "
2065 NEXT A
2070 FOR A=18 TO 4 STEP -2
2075 PRINT AT A,1;CHR$ T;AT A,19
;CHR$ T
2080 LET T=T+1
2085 NEXT A
2086 LET A=22
2087 PRINT AT 2,A;C$
2089 PRINT AT 11,A;S$
2091 PRINT AT 16,A;Z;"  .ZUG"
2092 PRINT AT 18,A;"ES ZIEHT"
2093 PRINT AT 20,A;S$
2094 PRINT AT 17,21;"
"
2095 PRINT AT 10,21;"
"
2096 PRINT AT 15,21;"
"
2099 RETURN
3000 SAVE "REVERS█"
3005 PRINT "█=SPIELERSTEIN C=COM
PUTERSTEIN"
3015 PRINT ,,"EINGABE DER ZUEGE=
BUCHST.,ZAHL"
3016 PRINT ,,"ZWISCHEN DEM ZU SE
TZENDEN STEIN UND DEM NAECHSTEN
EIGENEN STEIN MUSS MINDESTENS EI
N COMPUTER=    STEIN LIEGEN"
3020 PRINT ,,"BEI EINGABE EINES
FALSCHEN ZUGESN.L.DRUECKEN"
3025 PRINT ,,"SOLL DER COMPUTER
ANFANGEN,""0"" EINGEBEN"
3030 PRINT ,,"(AUCH WENN MAN SEL
BST NICHT MEHRZIEHEN KANN,""0""
EINGEBEN)"
3035 PRINT ,,,,"       A L L E S
  K L A R    ?"
3040 PRINT ,,"N.L. DRUECKEN"
3050 INPUT N$
3055 CLS
3060 RUN
```

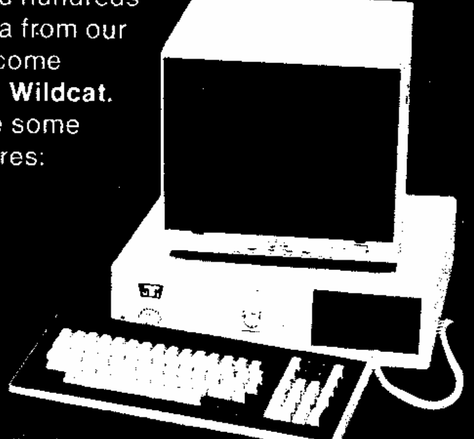
WILDCAT™

Looks Like an IBM™ Works Like an Apple™

To a land where fruit and flowers reign supreme, comes the awesome power of the **Wildcat** to challenge their supremacy.

Wildcat is a sleek styled mobile computer designed for your vehicle, boat or aircraft. It looks like an IBM PC™ and is fully software compatible with the Apple II™ product line at a list price that would put a smile on the face of the most frugal computer buyer.

But price isn't everything. All those features that would cost you hundreds of dollars extra from our competitors, come standard with **Wildcat.** Let's compare some of these features:



	Apple IIe	Wildcat
Detachable keyboard	No	STD
Full numeric key pad	Option	STD
Full functions keys	No	41
Built in disk controller	No	STD
Parallel printer port	No	STD
RS 232 serial port	No	STD
Game port	1	2
RGB video out	Option	STD
Composite video	STD	STD
RF video for TV	Option	STD
CP/M	Option	STD
Hi Res graphics (6 color)	STD	STD
Low Res graphics (16 color)	STD	STD
64KB memory	STD	STD
Half high disk drives	No	STD
Converters for vehicles, boats, and aircraft	No	Option
Aluminum carrying case	No	Option
List price	$1940*	$1099.00*

For more information on the all new **Wildcat**, see your local computer dealer or call or write:



COMPUTER AND PERIPHERAL PRODUCTS
1530 S. Sinclair
Anaheim, CA 92806
(714) 978-9820

*Computer plus one disk drive

Warning: This equipment is exempt from compliance with FCC testing requirements pursuant to 47 CFR 15.801 (c) (1)
Operation of this equipment in a residential area may cause interference.

CIRCLE 181 ON READER SERVICE CARD

BASIC-Programm zur Auswertung von Funktionen und Folgen

Dieses in BASIC geschriebene Programm für den Dragon 32 ermöglicht es seinem Anwender jede nur denkbare Folge oder Funktion sowohl graphisch (mit Hilfe eines Koordinatenkreuzes) als auch rechnerisch (mit Hilfe von ausgegebenen Zahlenkolonnen) darzustellen und auszuwerten.

Hier noch einige Tips und Anmerkungen zu dem sich weitgehendst selbsterklärenden Programm:
Nach dem starten des Programmes werden Sie zuerst daran erinnert eine der von Ihnen gewünschten Funktionen dem Programm zur Grundlage zu setzen, indem Sie es anhalten, und durch EDIT 590 diese dort einsetzen.

Dieses brauchen Sie jedoch zunächst nicht beachten da im Programm die Beispielfunktion Y=SINUS X eingegeben ist. Um eine richtige Eingabe kontrollieren zu können, fahren Sie also direkt fort.
Sie können nun zwischen A GRAPHISCHER und B RECHNERISCHER Auswertung wählen!

In beiden Fällen werden Sie auf die Möglichkeit hingewiesen, am Ende des Programmablaufes durch drücken der ↑ Taste neue, vielleicht bessere Werte vorauszusetzen um ein optimaleres Ergebnis zu erlangen.
Zu diesen Werten gehören:
(bei A und B) die Festlegung des Zahlenbereiches der für x eingesetzt werden soll: Dieses geschieht durch Eingabe der ERSTEN UND DER LETZTEN Zahl dieses Bereiches sowie die Bestimmung des Abstandes den die Werte in diesem Bereich zueinander haben sollen.
z. B.: erster Wert = −2
letzter Wert = +2
Abstand = 0.5
So lauten die Zahlen, die nacheinander für x eingesetzt werden:
−2/−1.5/−1/−0.5/0/+0.5/+1/+1.5/+2
(Nur bei A): Die Festlegung der Einheit auf der x-Achse sowie auf der y-Achse: Dieses geschieht durch die Festlegung des Wertes den die Teilstriche auf dem Koordinatenkreuz ha-

weiter Seite 17

```
10 CLS0
20 PRINT
30 REM ************************
40 REM BASIC PROGRAMM ZUR
50 REM RECHNERISCHEN UND
60 REM GRAPHISCHEN AUSWERTUNG
70 REM VON BELIBIGEN FUNKTIONEN
80 REM        UND FOLGEN
90 REM -RAGON 32 AUSFUEHRUNG-
100 REM
110 REM -COPYRIGHT 1984 BY:
120 REM    MICHAEL BRUST
130 REM -MOERSERSTRASSE 180-
140 REM -4100 DUISBURG 17  -
150 REM ************************
160 PRINT"**BASIC-PROGRAMM ZUR AUSWERTUNG****VON folgen UND funktionen***" "
170 FORI=1TO2
180 PLAY"T15CDEFGGAAAGAAAGFFFEEDDDC"
190 NEXTI
200 CLS0
210 PRINT:PRINT
220 PRINT"*****ist*die*zu*behandelnde****************funktion**************noch
*nicht*eingegeben*so*ist***sie*durchEDITin590einzusetzen!*"
230 PRINT"ACHTUNG! DANACH IST DAS PROGRAMM       ERNEUT ZU STARTEN!
240 FORI=1TO3
250 PLAY"T15CDEFGGAAAAGAAAAGFFFFEEDDDDC"
260 NEXT I
270 CLS0
280 PLAY"T255ACDF"
290 PRINT
300 PRINT"WAS WUENSCHEN SIE "
310 PRINT"    A) GRAPHISCHE "
320 PRINT"    B) RECHNERISCHE "
330 PRINT"DARSTELLUNG DER FUNKTION?"
340 PRINT:INPUT" A ODER B ?>>>";WA$
350 PMODE3,1:SCREEN1,1:PCLS
360 CLS0
370 PLAY"T255CCDDEEFFEEDDCCC"
380 PRINT"NACH DEM ENDE KANN DAS PROGRAMM DURCH DRUECKEN DER -^- TASTE    WIEDER
 GESTARTET WERDEN UM EVTL.ANDERE WERTE AUSZUPROBIEREN!
390 IFWA$="A"THEN410
400 PRINT:INPUT"WIE SOLL DIE AUSGABE DER WERTE  ERFOLGEN?
  A) SCHNELL                          B) LANGSAM                        C) ZUM MITSCHR
EIBEN          ";M$
410 PRINT"BITTE GEBEN SIE DIE JEWEILS     GEWUENSCHTEN WERTE EIN!"
420 PRINT
```



```
430 PRINT"DIE FUER X EINZUSETZENDEN ZAHLEN:"
440 INPUT"DER ERSTE EINZUSETZENDE WERT    (NORMAL:-10)SEI:";R
450 INPUT "DER LETZTE EINZUSETZENDE WERT   (NORMAL:+10)SEI:";D
460 SOUND6,1
470 CLS0
480 PRINT
490 INPUT"DER ABSTAND DER FUER X EINZU-   SETZENDEN ZAHLEN ZUEINANDER     (NORMA
L:0.5) SEI:";P
500 SOUND60,1
510 IFWA$="B"THEN550
520 PRINT
530 INPUT "DIE EINHEIT AUF DER X ACHSE     (NORMAL:1) SEI:";J
540 INPUT"EINHEIT AUF DER Y ACHSE         (NORMAL:1) SEI:";K
550 SOUND100,3
560 FOR X=R TO D STEP P
570 REM IN 590 Y=F(X) IST DIE       BEISPIELFUNKTION Y=SIN(X)      DURCH DIE GEWU
ENSCHTE ZU        ERSETZEN!!                        Z.B. Y=X+5 O. Y=(1/9*X)^X
580 REM************************
590 Y=SIN(X)
600 REM************************
610 IF WA$="A"THEN630
620 IFWA$="B"THEN840ELSE370
630 A=126+(X*((1/J)*12))
640 B=95-(Y*((1/K)*8))
650 PLAY"T255A"
660 REM
670 REM***KOORDINATENSYSTEM:***
680 PMODE 4,1 :SCREEN 1,1
690 REM DURCH AENDERUNG DER KOORDINATE 126,95 IN710;720;730;740 WIRD DIE LAGE DE
S SYSTEMMITTELPUNKTES (0-PUNKT) VERSCHOBEN
700 REM
710 DRAW" BM 126,95 ; S4 ; U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R
2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U8L2R2U
8L2R2U8L2R2U8L2R2U8L2R2U8L2R4"
720 DRAW" BM 126,95 ;D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R
2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D8L2R2D
8L2R2D8L2R2D8L2R2D8L2R4"
730 DRAW" BM 126,95 ;L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12
D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U
2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2L12D2U2"
740 DRAW" BM 126,95 ;R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12
D2U2R12D2U2R12D2U2R122U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2R12D2U2
R12D2U2R12D2U2R12D2U2"
750 IF A>250-PTHEN GOTO810
760 IF A<0THEN NEXT X
770 IF B>188THEN NEXT X
780 IF B<0 THEN NEXT X
790 LINE(A,B)-(A,B),PSET
800 NEXT X
810 A$=INKEY$
815 IFA$="^"THENCLS0
820 IF A$="^"THEN300ELSE810
830 REM****RECHNERISCH********
840 PRINT"X=";X,"Y=";Y
850 IFM$="B"THENFORS=1TO40
860 IFM$="C"THENFORS=1TO800
870 IFM$="B"ORM$="C"THENNEXTS
880 NEXT X
890 A$=INKEY$
895 IFA$="^"THENCLS0
900 IFA$="^"THEN270ELSE890
910 REM ende
```

ben sollen. Dieser ist normal 1 (die Teilstriche erhalten den Wert 1/2/3/4 usw.). Will man aber besondere Kurven darstellen
(z. B. besonders enge) so empfiehlt es sich den Wert des Teilstrichabstandes zu verändern. Z. B. eine Verkleinerung auf 0.2 (Teilstriche 0.2/0.4/0.6 usw.). Geht eine Kurve zu sehr aus dem dargestellten Bereich heraus, so sollte man den Wert auf etwa 10 vergrößern (Teilstriche 10/20/30 usw.). Sie können selbstverständlich auch unterschiedliche Einheiten für die x- und y-Achse wählen! Je nach den Anforderungen die die Funktion stellt, sollte der Zahlenbereich und der Abstand entsprechend verändert werden. Setzt man zu extreme Werte ein, so kann es zu einem FC bzw. SN ERROR kommen. Achten Sie auch darauf, keine Division durch Null einzugeben.

The Big Quest



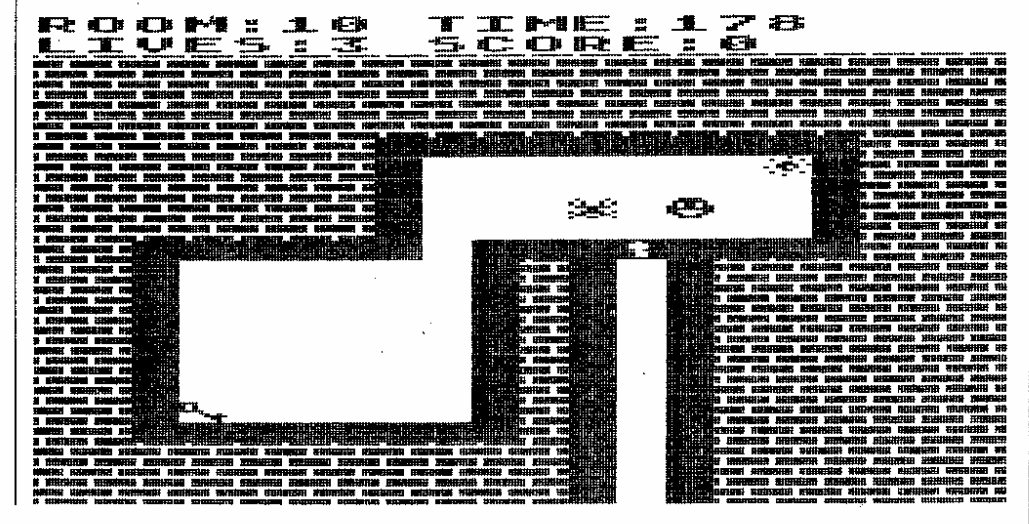
Das hier abgedruckte Programm ist auf allen Atari-Computern lauffähig

Sie übernehmen die Rolle eines sehr mutigen Abenteurers, der es sich in den Kopf gesetzt hat, das Schloß des Grafen Frank von Stein zu plündern.

Doch damit ist Herr von Stein gar nicht einverstanden und hetzt Ihnen allerlei Monster und Gewürm auf den Hals. Außerdem sind die Räume des Schlosses mit einer Schließvorrichtung versehen, welche die Türen automatisch schließt, sobald jemand den Raum betritt. Diese Türen können Sie nur wieder öffnen, wenn Sie sich den Schlüssel holen, der irgendwo im Raum herumliegt. Und noch etwas: Wenn Sie zulange in dem Raum bleiben, kann Herr von Stein Ihre genaue Position ausmachen und den Raum einfach mit Wasser füllen. Also Beeilung!! Sonst setzt's was!! Sie steuern den kleinen (noch) grinsenden Mann mit dem Joystick in alle acht Richtungen und müssen nun versuchen, die Perle und den Schlüssel zu klauen (erst die Perle, dann den Schlüssel!!), um sich dann so schnell wie möglich zu verdünnisieren. Die Wände sind übrigens elektrisch geladen. (Kicher). Wenn Sie von einem Monster bedroht werden, können Sie sich dessen erwehren, indem Sie schnell den Kopf (auch Trigger genannt) drücken, um das Monster mit einem lauten Knall zu „zappen". Doch Achtung: Das funktioniert nur nach rechts!! Da das „Zappen" sehr laut ist, kann Herr von Stein Sie jetzt natürlich viel leichter hören ... Es gibt verschiedene Monster mit verschiedenen Eigenschaften. Das eine sieht aus wie ein Geist und hat Angst vor Wänden, das andere sieht aus wie eine Spinne und schwebt durch die Wand ...

Wenn Sie einen Raum heil verlassen haben, werden Sie sogleich in den nächsten befördert (der natürlich schwieriger ist). Wieviele Räume es gibt??? Das müssen Sie selbst herausfinden ...

Viel Spaß!!




```
1 REM *****************
2 REM * THE BIG QUEST! *
3 REM * BY KEMAL EZCAN *
4 REM * COPYRIGHT 1984 *
5 REM *****************
10 PAGE=PEEK(106)-8:CHS=PAGE*256:POKE 559,0:POKE 77,128:RESTORE 28000
12 FOR I=128 TO 471:POKE CHS+I,PEEK(57344+I):NEXT I
14 READ C:IF C<>-1 THEN FOR I=0 TO 7:READ A:POKE CHS+C*8+I,A:NEXT I:GOTO 14
16 POKE 559,34:POKE 77,0:GOTO 6000
50 REM START DER SUCHE
51 SOUND 0,0,0,0
60 P=0:M=3:ROOM=1:MAXROOM=10
70 GRAPHICS 17:POKE 756,PAGE:POKE 708,15:POKE 709,24
80 ? #6;"ROOM:1  TIME:000":? #6;"LIVES:3 SCORE:0"
100 REM NEUER RAUM/LEBEN VERLOREN
110 GOSUB 1000
120 TIME=200:MCTR=1:RAUSFLG=0:H=32
130 POSITION 5,0:? #6;ROOM
150 REM MAIN LOOP
160 TIME=TIME-1:POSITION 13,0:? #6;TIME;" ":IF TIME=0 THEN 4500
170 ST=STICK(0):XR=0:YR=0
180 XR=XR-(ST>8 AND ST<12):XR=XR+(ST>4 AND ST<8):YR=YR-(ST=10 OR ST=6 OR ST=14):
YR=YR+(ST=9 OR ST=5 OR ST=13)
190 IF XR=0 AND YR=0 THEN 250
195 IF X+XR>19 OR X+XR<0 OR Y+YR<2 OR Y+YR>23 THEN 7000
200 LOCATE X+XR,Y+YR,Z:IF Z=168 OR Z=MCHR THEN 5000
210 IF Z=32 THEN SOUND 0,100,8,10:COLOR 32:PLOT X,Y:X=X+XR:Y=Y+YR:COLOR 33:PLOT
X,Y:SOUND 0,0,0,0
220 IF Z=41 THEN GOSUB 1200
```



```
 230 IF RAUSFLG=1 AND Z=42 THEN GOSUB 1300
 250 REM ENDE DER BEWEGUNG
 260 IF MFLG THEN MCTR=MCTR+1:IF MCTR>MSPD THEN MCTR=MCTR-MSPD:GOSUB 1400
 270 IF STRIG(0)=0 THEN GOSUB 1500
 990 GOTO 150:REM END OF MAIN LOOP
1000 REM SCREEN AUFBAU
1010 RESTORE 9900+ROOM*100
1020 READ X,Y,MX,MY,MCHR,WFLG,MSPD,MFLG,PX,PY,SX,SY,TX,TY,SE2,SE3
1030 FOR I=2 TO 23:COLOR 142:PLOT 0,I:DRAWTO 19,I:NEXT I:POKE 710,SE2:POKE 711,S
E3
1040 READ X1,Y1,X2,Y2:IF X1=-1 THEN 1060
1050 COLOR 32:FOR I=Y1 TO Y2:PLOT X1,I:DRAWTO X2,I:NEXT I:GOTO 1040
1060 READ X1,Y1,C:IF X1=-1 THEN 1080
1065 COLOR 168:IF C=2 THEN PLOT X1,Y1:GOTO 1060
1070 DRAWTO X1,Y1:GOTO 1060
1080 COLOR 33:PLOT X,Y:IF MFLG THEN COLOR MCHR:PLOT MX,MY
1090 COLOR 41:PLOT PX,PY:COLOR 42:PLOT SX,SY
1100 FOR W=1 TO 50:NEXT W
1110 COLOR 171:PLOT TX,TY:FOR I=15 TO 0 STEP -0.2:SOUND 0,60,4,I:NEXT I
1130 RETURN
1200 REM PERLE GENOMMEN
1210 P=P+50:POSITION 14,1:? #6;P:COLOR 32:PLOT X+XR,Y+YR
1220 FOR I=15 TO 0 STEP -0.6:SOUND 0,I/3+28,12,I:NEXT I
1230 RAUSFLG=1:RETURN
1300 REM SCHLUESSEL GENOMMEN
1310 COLOR 32:PLOT X+XR,Y+YR:P=P+20:POSITION 14,1:? #6;P
1320 FOR I=15 TO 0 STEP -0.4:SOUND 0,152,12,I:NEXT I:FOR W=1 TO 20:NEXT W
1330 FOR I=30 TO 10 STEP -0.5:SOUND 0,I,12,10:SOUND 1,I,4,10:SOUND 2,I,2,10:NEXT
 I:SOUND 0,0,0,0:SOUND 1,0,0,0
1340 SOUND 2,0,0,0:COLOR 32:PLOT TX,TY:RETURN
1400 REM MONSTER MOVE
1410 XR=0:YR=0
1420 XR=XR-(X<MX):XR=XR+(X>MX):YR=YR-(Y<MY):YR=YR+(Y>MY)
1430 LOCATE MX+XR,MY+YR,Z
1440 IF Z=32 OR Z=WFLG THEN COLOR H:PLOT MX,MY:MX=MX+XR:MY=MY+YR:COLOR MCHR:PLOT
 MX,MY:H=Z:RETURN
1450 IF Z=33 THEN 5000
1460 RETURN
1500 REM SHOOT
1510 TIME=TIME-10:IF TIME<0 THEN TIME=0
1520 POSITION 13,0:? #6;TIME:IF TIME=0 THEN 4500
1530 LOCATE X+1,Y,Z:IF Z<>32 THEN RETURN
1540 COLOR 45:PLOT X+1,Y
1550 FOR I=0 TO 150 STEP 10:SOUND 0,I,8,15-I/10:NEXT I
1560 LOCATE X+2,Y,Z:COLOR 32:PLOT X+1,Y:IF Z<>MCHR THEN RETURN
1570 FOR I=15 TO 0 STEP -1:SETCOLOR 1,RND(0)*16,I:SOUND 0,RND(0)*20,8,I:NEXT I:P
=P+25:POSITION 14,1:? #6;P
1580 MFLG=0:COLOR H:PLOT MX,MY:POKE 709,24:RETURN
4500 REM KAPUTT DURCH ZEIT
4510 POP
4520 FOR I=0 TO 15 STEP 0.2:SOUND 0,I/2,8,I:NEXT I
4530 IF MFLG THEN COLOR 32:PLOT MX,MY
4540 SETCOLOR 1,7,2:COLOR 8
4550 FOR I=23 TO 2 STEP -1:SOUND 0,250-I*10,8,15:PLOT 0,I:DRAWTO 19,I:FOR W=1 TO
 20:NEXT W:NEXT I
4560 FOR W=1 TO 100:NEXT W
4570 FOR I=2 TO 23:SOUND 0,200,8,23-I:COLOR 32:PLOT 0,I:DRAWTO 19,I:NEXT I
4580 GOTO 5030
5000 REM MANN VERLOREN
5005 POP :POP :POP
5010 FOR I=15 TO 0 STEP -0.5:SOUND 0,0,4,I:A=A*1:SOUND 0,5,4,I:NEXT I:SOUND 0,0,
```



```
0,0
5020 COLOR 32:PLOT X,Y
5030 M=M-1:POSITION 6,1:? #6;M
5040 IF M>0 THEN 100
5050 FOR U=150 TO 200 STEP 8:FOR I=U TO U-100 STEP -10:SOUND 0,I,10,10-(U-150)/5
:NEXT I:FOR W=1 TO 10:NEXT W:NEXT U
6000 GRAPHICS 0:POKE 752,1:POKE 82,0:POKE 712,52:POKE 710,116
6010 ? "DEINE PUNKTE:";P:? :?
6020 ? "        T H E   B I G   Q U E S T  !":? :? "          EIN SPIEL VON KEMAL EZC
AN!":? :? :?
6030 ? "        DRUECKE START UM ZU BEGINNEN":? :? :? :?
6040 ? "DIE AUFGABE BESTEHT DARIN, IN DEN VIELEN"
6050 ? "RAEUMEN  DES  GEHEIMNISVOLLEN  SCHLOSSES"
6060 ? "DIE VERSTECKTEN  PERLEN  ZU  FINDEN  UND"
6070 ? "DANN HEIL ZU ENTKOMMEN."
6080 L=12:T=1:RESTORE 6200
6090 IF PEEK(53279)=6 THEN GOTO 50
6095 READ T:IF T=-1 THEN RESTORE 6000:READ T:L=L-(L>0)
6100 SOUND 0,T,10,L
6105 FOR W=1 TO 10:NEXT W
6110 GOTO 6090
6200 DATA 162,136,108,81,68,53,40,33
6210 DATA 182,153,121,91,76,60,45,37,-1
7000 REM RAUM GESCHAFFT
7010 POP :P=P+TIME:POSITION 14,1:? #6;P
7020 FOR I=15000 TO 0 STEP -230:SOUND 0,I,10,I/1000:NEXT I
7030 ROOM=ROOM+1:IF ROOM<=MAXROOM THEN 100
7040 GRAPHICS 18:? #6;"   GRATULATION!!!   ":? #6:P=P+1000
7050 ? #6;"DU HAST  DAS RAETSEL"
7060 ? #6;"DES  GEHEIMNISVOLLEN"
7070 ? #6;"SCHLOSSES GELOEST!!!"
7080 FOR W=1 TO 600:NEXT W
7090 GOTO 6000

10000 DATA 16,17,0,0,0,0,0,0,4,6,10,10,17,17,52,116
10010 DATA 3,5,7,15,8,9,12,20,13,13,16,20,17,17,19,17,-1,0,0,0
10020 DATA 2,4,2,8,4,1,8,8,1,13,8,1,13,12,1,17,12,1,17,16,1,19,16,1,19,18,2,17,1
8,1,17,21,1,8,21,1
10030 DATA 8,16,1,2,16,1,2,4,1,-1,0,0
10100 DATA 3,15,8,12,3,0,5,1,4,10,7,15,2,15,36,68
10110 DATA 3,5,9,20,10,11,16,20,0,15,2,15,-1,0,0,0,10,10,2,17,10,1,17,21,1,2,21,
1,2,16,1,0,16,1,0,14,2,2,14,1
10120 DATA 2,4,1,10,4,1,10,10,1,6,8,2,6,16,1,14,16,1,-1,0,0
10200 DATA 14,13,0,0,0,0,0,0,4,7,8,8,14,12,22,52
10210 DATA 3,6,9,13,10,13,16,20,14,2,14,12,-1,0,0,0
10220 DATA 15,2,2,15,12,1,17,12,1,17,21,1,10,21,1,10,14,1,2,14,1,2,5,1,10,5,1,10
,12,1,13,12,1,13,2,1,-1,0,0,0
10300 DATA 8,18,13,12,4,168,3,1,8,13,8,7,8,19,4,36
10310 DATA 5,5,14,18,3,8,16,15,8,19,8,23,-1,0,0,0
10320 DATA 7,23,2,7,19,1,4,19,1,4,16,1,2,16,1,2,7,1,4,7,1,4,4,1,15,4,1,15,7,1,17
,7,1,17,16,1,15,16,1,15,19,1,9,19,1
10330 DATA 9,23,1,9,7,2,9,16,1,6,12,2,13,12,1,-1,0,0
10400 DATA 11,14,16,14,5,0,3,1,5,18,4,9,11,15,244,4
10410 DATA 2,7,7,19,8,12,17,14,15,12,17,20,11,15,11,23,-1,0,0,0
10420 DATA 10,23,2,10,15,1,8,15,1,8,19,1,1,19,1,1,6,1,8,6,1,8,11,1,18,11,1,18,21
,1,14,21,1,14,15,1,12,15,1,12,23,1
10430 DATA -1,0,0
10500 DATA 9,21,9,5,6,168,2.5,1,3,7,16,7,9,22,196,244
10510 DATA 2,4,17,21,9,22,9,23,-1,0,0,0
10520 DATA 8,23,2,8,22,1,1,22,1,1,3,1,18,3,1,18,22,1,10,22,1,10,23,1,1,9,2,5,9,1
,5,6,1,14,6,2,14,9,1,18,9,1,-1,0,0
10600 DATA 9,5,9,18,7,168,1.5,1,8,20,10,20,9,4,166,196
10610 DATA 2,5,17,20,9,4,9,6,9,2,9,4,-1,0,0,0
```




```
10620 DATA 8,2,2,8,4,1,1,4,1,1,21,1,18,21,1,18,4,1,10,4,1,10,2,1,3,6,2,3,19,1,16
,19,1,16,6,1,9,20,2,-1,0,0
10700 DATA 17,19,0,0,0,0,0,0,2,20,16,5,18,19,150,164
10710 DATA 18,19,19,19,2,4,17,21,-1,0,0,0
10720 DATA 1,3,2,18,3,1,18,22,1,1,22,1,1,3,1,16,18,1,16,19,2,2,5,1,7,5,2,17,15,1
,8,5,2,17,14,1,12,4,2,16,8,1,11,4,2
10730 DATA 16,9,1,3,11,2,13,21,1,4,11,2,14,21,1,-1,0,0
10800 DATA 17,21,15,7,5,0,1,1,17,15,17,4,18,21,184,116
10810 DATA 2,4,17,21,18,21,19,21,-1,0,0,0
10820 DATA 1,3,2,18,3,1,18,22,1,1,22,1,1,3,1,3,16,2,18,16,1,1,18,2,16,18,1,3,20,
2,18,20,1,-1,0,0
10900 DATA 12,10,4,15,6,0,1.8,1,15,7,3,19,12,11,246,68
10910 DATA 3,11,8,19,8,7,15,10,12,11,12,23,-1,0,0,0
10920 DATA 11,23,2,11,11,1,9,11,1,9,20,1,2,20,1,2,11,1,7,11,1,7,6,1,16,6,1,16,11
,1,13,11,1,13,23,1,-1,0,0
28000 DATA 1,60,126,90,219,255,195,102,60
28010 DATA 3,60,90,90,255,219,165,255,165
28020 DATA 4,126,42,221,123,30,56,110,66
28030 DATA 5,40,84,186,84,124,198,186,238
28040 DATA 6,66,165,165,126,24,189,102,153
28050 DATA 7,98,181,173,126,195,189,247,110
28060 DATA 8,255,255,255,255,255,255,255,255
28070 DATA 9,8,66,24,61,180,24,66,16
28080 DATA 10,224,160,160,240,26,15,6,2
28090 DATA 11,255,239,199,199,239,231,231,255
28100 DATA 13,0,0,0,119,221,0,0,0
28110 DATA 14,191,0,251,251,251,0,191,191,-1
```

Fünf gewinnt

Fünf gewinnt ist ein Strategiespiel für 2 Personen mit 2 Joysticks, geschrieben in ATARI Basic.

Taktisches Gespür, strategisches und logisches Denken, schnelles Erkennen der Situationen sind gefragt. – Eine Version des beliebten Strategiespieles. Zu Beginn des Spieles wird eine Spielanleitung angeboten, die das Programm kurz beschreibt.
Der Sinn des Spieles ist relativ einfach: Wer es als erster schafft, 5 Steine in eine Reihe zu bekommen, horizontal, vertikal oder diagonal, gewinnt das Spiel.
Für Spieler 1 (blau) ist am Anfang des Spieles der 1. Stein gesetzt.
Nach Ende des Spieles wird der Verlierer getröstet und durch drücken der Start-Taste das Spiel von neuem begonnen.
Also dann: Viel Spaß!

```
100 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
110 REM @@                              @@
120 REM @@ 5 GEWINNT FUER ATARI 800     @@
130 REM @@                              @@
140 REM @@           VON                @@
150 REM @@                              @@
160 REM @@       ELLMAR MEDER           @@
170 REM @@                              @@
180 REM @@    (C) COPYRIGHT 1984        @@
190 REM @@                              @@
200 REM @@                              @@
210 REM @@                              @@
220 REM @@       LAENGE: 7.234 KBYTE    @@
230 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
240 POKE 82,0
250 FOR I=1 TO 3:GOSUB 460
260 FOR PSE=1 TO 100:NEXT PSE
270 FOR TD=1 TO 7:FOR K=3 TO 10
280 SOUND 1,K,10,8
290 NEXT K:NEXT TD
300 NEXT I
310 SOUND 1,0,0,0
320 GOTO 450
330 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
340 REM @@ UNTERPROGRAM ZUM FEST-      @@
350 REM @@ STELLEN OB KONSOLEN-        @@
360 REM @@ TASTE GEDRUECKT             @@
370 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
380 POKE 53279,8
```



```
390 TRAP 390:IF PEEK(53279)<>6 OR PEEK(53279)=7 THEN GOTO 390
400 RETURN
410 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
420 REM @@ UNTERPROGRAMM ZUR DAR-    @@
430 REM @@ STELLUNG DES P.ANFANGS    @@
440 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
450 POKE 755,1:GOTO 600
460 GRAPHICS 5
470 SETCOLOR 4,0,0:SETCOLOR 2,0,0
480 SETCOLOR 1,13,9:COLOR 2
490 PLOT 50,8:DRAWTO 30,8
500 DRAWTO 30,19:DRAWTO 47,19
510 DRAWTO 47,28:DRAWTO 30,28
520 DRAWTO 30,31:DRAWTO 50,31
530 DRAWTO 50,16:PLOT 50,16
540 DRAWTO 33,16:DRAWTO 33,11
550 DRAWTO 50,11:DRAWTO 50,8
560 PLOT 0,0:DRAWTO 79,0
570 DRAWTO 79,39:DRAWTO 0,39
580 DRAWTO 0,0
590 RETURN
600 PRINT "(12 SPACES) 5 G E W I N N T"
610 PRINT "(18 SPACES)       VON(20 SPACES)        ";
620 PRINT "(13 SPACES)  ELLMAR MEDER";
630 PRINT "(25 SPACES)              (C) 1984";
640 FOR PSE=1 TO 1000:NEXT PSE
650 DIM ANTW$(1)
660 PRINT "}Wird Spielanleitung benoetigt (J/N) ";
670 INPUT ANTW$:IF ANTW$="N" THEN GOTO 860
680 GRAPHICS 0:SETCOLOR 2,0,0
720 PRINT "Dies ist ein Spiel fuer 2 Personen.":PRINT
730 PRINT "Der Sinn des Spieles besteht darin,dass":PRINT
740 PRINT "man versuchen muss 5 Steine entweder in":PRINT
750 PRINT " HORIZONTALER, VERTIKALER  ODER IN DIA-":PRINT
760 PRINT "GONALER Reihenfolge zusammenzusetzten.":PRINT
770 PRINT "Durch druecken des roten Knopfes am":PRINT
780 PRINT " Joystick(1/2) wird der Stein gesetzt.":PRINT
790 PRINT "Fuer Spieler 1 (blau) ist der erste ":PRINT
800 PRINT "Stein zu Beginn des Spieles gesetzt.":PRINT
810 PRINT "VIEL SPASS                   --->START";
820 GOSUB 380
830 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
840 REM @@BEGINN DES HAUPTPROGRAMMS  @@
850 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
860 CLR :POKE 12,119:POKE 13,228:POKE 755,1:POKE 82,0
870 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
880 REM @@@     INITIALISIERUNG      @@@
890 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
900 GOSUB 1800
910 PLR=1-PLR
920 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
930 REM @@ FESTSTELLUNG DER GEG-     @@
940 REM @@ WERTIGEN POSITION         @@
950 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
960 PK=PEEK(PPOS(PLR))
970 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
980 REM @@  SETZEN DER STEINE        @@
990 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1000 POKE PPOS(PLR),PIECE(PLR)
```

```
1010 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1020 REM @ BLINKDAUER DES STEINCHENS @@
1030 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1040 FOR BLD=1 TO 10:NEXT BLD
1050 ST=STICK(PLR):IF STRIG(PLR)=0 AND PK=83 THEN 1170
1060 POKE PPOS(PLR),PK-128*(PK>128)
1070 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1080 REM @ WENN STICK NICHT BEWEGT   @@
1090 REM @ BLEIBT POSITION ERHALTEN  @@
1100 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1110 IF ST=15 THEN 1000
1120 POKE PPOS(PLR),PK
1130 PPOS(PLR)=PPOS(PLR)+DIR(ST)
1140 PK=PEEK(PPOS(PLR))
1150 IF PK<>83 AND PK<>PIECE(0) AND PK<>PIECE(1)
     THEN PPOS(PLR)=PPOS(PLR)-DIR(ST
)
1160 GOTO 960
1170 POKE PPOS(PLR),PIECE(PLR)
1180 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1190 REM @@ UEBERPRUEFT OB GENAU 5   @@
1200 REM @@ STEINE IN EINER REIHE    @@
1210 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1220 FOR I=5 TO 14
1230 TEMP=PPOS(PLR):DIR=DIR(I):N=0
1240 IF DIR=0 THEN 1310
1250 TEMP=TEMP+DIR
1260 IF PEEK(TEMP)=PIECE(PLR) THEN N=N+1:GOTO 1250
1270 N=0
1280 TEMP=TEMP-DIR
1290 IF PEEK(TEMP)=PIECE(PLR) THEN N=N+1:GOTO 1280
1300 IF N=5 THEN 1360
1310 NEXT I
1320 GOTO 910
1330 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1340 REM @ UEBERPRUEFT WER GEWONNEN  @@
1350 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1360 FOR I=1 TO 100
1370 POKE 712,PEEK(53770)
1380 NEXT I
1390 POKE 712,0
1400 P=PEEK(710+PLR):C=INT(P/16)
1410 FOR I=1 TO 10
1420 POKE 710+PLR,0
1430 SOUND 0,180,9,8
1440 FOR PSE=1 TO 10:NEXT PSE
1450 POKE 710+PLR,P
1460 SOUND 0,100,10,8
1470 FOR PSE=1 TO 20:NEXT PSE
1480 NEXT I
1490 P=PEEK(711-PLR):C=INT(P/16)
1500 FOR I=14 TO 0 STEP -0.2
1510 SOUND 0,I*2,12,I
1520 POKE 711-PLR,C*16+I
1530 NEXT I
1540 POKE 711-PLR,0
1550 FOR I=1 TO 10
1560 POKE 53274,PEEK(53770)
1570 NEXT I
```




```
1580 FOR PSE=1 TO 500:NEXT PSE
1590 GRAPHICS 17:SETCOLOR 0,0,0
1600 IF PLR THEN GOTO 1680
1610 POSITION 2,1:? #6;"sPieler eins ist":POSITION 6,7:? #6;"*sie9er*"
1620 POSITION 6,6:? #6;"********":POSITION 6,8:? #6;"********"
1630 POSITION 0,12:? #6;"PECH SPIELER ZWEI,"
1640 POSITION 0,13:? #6;"VIELLEICHT KLAPPT"
1650 POSITION 0,14:? #6;"ES DAS NAECHSTE"
1660 POSITION 0,15:? #6;"MAL."
1670 GOTO 1740
1680 POSITION 2,1:? #6;"SPIELER ZWEI IST":POSITION 6,7:? #6;"*SIEGER*"
1690 POSITION 6,6:? #6;"@@@@@@@@":POSITION 6,8:? #6;"@@@@@@@@"
1700 POSITION 0,12:? #6;"PECH SPIELER EINS."
1710 POSITION 0,13:? #6;"VIELLEICHT KLAPPT"
1720 POSITION 0,14:? #6;"ES DAS NAECHSTE"
1730 POSITION 0,15:? #6;"MAL."
1740 FOR PSE=1 TO 300:NEXT PSE
1750 POSITION 0,22:? #6;"DRUECKE START FUER ":POSITION 4,23:? #6;
     "NEUES SPIEL"
1760 FOR PSE=1 TO 300:NEXT PSE
1770 GOSUB 380
1780 GOTO 860
1790 GOTO 1790
1800 DIM DIR(15)
1810 FOR I=5 TO 15
1820 READ A:DIR(I)=A
1830 NEXT I
1840 DATA 21,-19,1,0,19,-21,-1,0,20,-20,0
1850 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1860 REM @@  BILDSCHIRM AUFBAUEN      @@
1870 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
1880 GRAPHICS 16+1:SETCOLOR 0,0,0
1890 POKE 756,226
1900 SCR=PEEK(88)+256*PEEK(89)
1910 FOR I=2 TO 20
1920 FOR J=2 TO 17
1930 POKE SCR+I*20+J,83
1940 NEXT J
1950 NEXT I
1960 FOR I=2 TO 17
1970 POKE SCR+40+I,87
1980 POKE SCR+420+I,88
1990 NEXT I
2000 FOR I=2 TO 20
2010 POKE SCR+20*I+2,65
2020 POKE SCR+20*I+17,68
2030 NEXT I
2040 POKE SCR+42,81:POKE SCR+57,69
2050 POKE SCR+422,90:POKE SCR+437,67
2060 POSITION 4,0:? #6;"FUENF GEWINNT"
2062 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
2064 REM @@ AUFBAU DES STEINCHENS     @@
2066 REM @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
2070 DIM PPOS(1),PIECE(1),COUNT(1)
2080 PPOS(0)=SCR+229:PPOS(1)=SCR+209
2090 PIECE(0)=148:PIECE(1)=212
2100 COUNT(0)=0:COUNT(1)=0
2110 POKE PPOS(0),PIECE(0)
2120 RETURN
```



SILVERSPAR

Die Erde wird von einem Weltraummonster, dem bösen Göört bedroht.

Dieser böse Göört kann jedoch nicht ungesehen die Erde erreichen, da die Radaranlagen ihn sofort erkennen würden, und er keine Chance hätte, zur Erde vorzudringen.
Aus diesem Grund schickt er Wellen von Ufos zur Erde, um die Radaranlagen zu vernichten.
Wenn es den UFOs gelingt, alle fünf Radartürme zu zerstören, so kann der böse Göört zur Erde vordringen.
Es liegt nun an Ihnen, diese Invasion zu verhindern. Schießen Sie die UFOs mit Ihrem Abfangjäger SILVERSPAR ab, bevor diese sämtliche Radartürme vernichtet haben.
Sollte Ihnen dies nicht gelingen, so müssen Sie gegen den bösen Göört höchstpersönlich antreten und mit ihm kämpfen.

SILVERSPAR ist ein Spiel für den TI 99/4 A, welches in Extended Basic geschrieben wurde und mittels Joystick 1 gesteuert wird.

SILVERSPAR ist eine Herausforderung an Ihr Reaktionsvermögen.
Tippen Sie das Listing ein, doch bevor Sie das Programm mit „RUN" starten, schreiben Sie dieses auf Kassette oder Diskette weg.
Das Programm besitzt einen Copyrightschutz, der beim Weglassen oder Verändern der Copyrightzeilen das System „abstürzen" läßt. Der Rechner kann dann nur noch mit <FCTN> <QUIT> oder ausschalten neu initialisiert werden.
Nach dem Programmstart mit „RUN" dauert es einige Sekunden, bis der Rechner die interne STACK-Liste erstellt hat. Jetzt wird der Bildschirm für ca. 30 Sekunden schwarz. Das Titelbild erscheint und die Titelmelodie wird gespielt.
Zum Spielstart drücken Sie den Aktionsknopf am Joystick 1. 10 Sekunden später beginnt das Spiel
Am unteren Bildschirmrand werden jetzt die Kirche, das Haus und ein Radarturm aufgebaut. Der SILVERSPAR erscheint am rechten Bildschirmrand. Steuern Sie mit dem Joystick den SILVERSPAR nach rechts, links, oben und unten. Der SILVERSPAR bewegt sich in Spielszene 1 nur sehr langsam in horizontaler Richtung. Dabei bewegt sich die Stadt jedoch relativ schnell in entgegengesetzter Richtung, so daß der Eindruck eines schnellen Fluges entsteht.
Gleichzeitig mit dem SILVERSPAR wird ein UFO auf dem Bildschirm sichtbar, welches es gilt abzuschießen. SILVERSPAR kann nur horizontal schießen. Um einen Schuß abzufeuern, muß der Aktionsknopf gedrückt und gleichzeitig der Joystick in die gewünschte Richtung bewegt werden (rechts oder links).
Die UFOs wechseln laufend die Farbe. Je nachdem, welche Farbe das UFO beim Abschuß hatte, erhalten Sie
UFO Typ 1: 10 bis 100 Punkte
UFO Typ 2: 20 bis 200 Punkte
UFO Typ 3: 30 bis 300 Punkte
UFO Typ 4: 40 bis 400 Punkte
Die UFOs versuchen, die Radartürme abzuschießen. Wird ein Radarturm getroffen, so wird dieser viermal durch einen neuen Turm ersetzt.
Nach 10 abgeschossenen UFOs ist ein leises, dann immer lauter werdendes „Schnarren" zu hören.
Dies ist das Zeichen, daß eine neue Welle UFOs naht.
Gelingt es Ihnen in einer Welle alle 10 UFOs abzuschießen, ohne daß hierbei ein Radarturm abgeschossen wurde, so erhalten Sie zu Ihren anfänglichen 5 SILVERSPARs einen zusätzlichen Abfangjäger hinzu.
Kollidiert SILVERSPAR jedoch mit einem UFO, so verliert er ein „Leben".
Wenn die UFOs alle 5 Radartürme zerstört haben, so erlischt die Stadt und eine Sirene wird lautstark hörbar. Der böse Göört kommt jetzt ins Bild.
Um eine neue Stadt mit weiteren 5 Radartürmen zu bekommen, müssen Sie den bösen Göört besiegen. Es genügt jedoch nicht, Göört mit einem einzigen Treffer zu erledigen. Göört muß zehnmal getroffen werden, bevor er vernichtet ist.
Mit jedem Treffer wechselt der böse Göört die Farbe und Sie erhalten, je nach Farbe, 50 bis 500 Punkte. Wenn Sie Göört zehnmal getroffen haben, so explodiert er und Sie erhalten eine neue Stadt.
Von Zeit zu Zeit schleudert Göört einen Feuerwirbel gegen den SILVERSPAR. Diesem Wirbel gilt es auszuweichen, denn eine Berührung mit dem Feuerwirbel oder mit dem bösen Göört kostet ein „Leben".
Das Spiel ist dann zu Ende, wenn alle SILVERSPARs zerstört sind.

Doch jetzt genug der Erklärungen. Ich wünsche ein ERRORfreies Eintippen und viel Spaß beim Spiel. Nachfolgend noch einige Tips zum Eintippen:

1) Einige Zeilen sind länger, als es der TI zuläßt. Tippen Sie die Zeile ein und schicken Sie diese mit <ENTER> ab. Danach holen Sie die Zeile mit <FCTN><REDO> wieder zurück und können den Rest eingeben.
2) Benutzer eines Diskettenlaufwerkes ohne Speichererweiterung sollten das Spiel bei abgeschalteter Peripheriebox eintippen und über einen Kassettenrecorder abspeichern, da durch den Dateibuffer des Laufwerkes der Speicher beim Programmlauf zu klein ist.
3) Wenn Sie ein Laufwerk mit Erweiterung besitzen, so ist vor dem Einlesen des Programmes von Kassette die Befehlsfolge: CALL FILES(1) <ENTER> NEW <ENTER> einzugeben, damit das Programm von Kassette eingelesen werden kann.
4) Während des Spielablaufes darf die <ALPHA LOCK>-Taste nicht eingerastet sein, da sonst der Joystick nicht richtig abgefragt wird.
5) Tippen Sie vor allem die DATA-Zeilen sehr genau ab, da ein hieraus resultierender Fehler sehr schwer zu finden ist.

```
100 REM ********************************
110 REM *                              *
120 REM *      S I L V E R S P A R     *
130 REM *------------------------------*
140 REM * ein Spiel fuer den TI 99/4 A *
150 REM * geschrieben in Extended BASIC *
160 REM * Steuerung mittels Joystick 1 *
170 REM *------------------------------*
180 REM * (c) 1984 by Volker Becker    *
190 REM *                              *
200 REM ********************************
210 REM
220 REM
230 ON ERROR 1600
240 CPR$="@ 1984 VOLKER BECKER"
250 SKP=ASC(SEG$(CPR$,8,1))-ASC(SEG$(CPR$,20,1))-4
260 IF SKP THEN 1600 ELSE 920
270 RANDOMIZE
280 ON ERROR STOP
290 DATA 104,112,120,124,136,140
300 OPTION BASE 1
310 DIM C(6),CU(4,2),B(6)
320 RESTORE 290
330 FOR I=1 TO 6 :: READ C(I):: NEXT I
```

```
340 CALL CLEAR :: CALL SCREEN(2):: CALL GRAPH
350 RESTORE 510
360 FOR I=1 TO 10 :: READ COL(I):: NEXT I
370 CALL HCHAR(2,1,30,704)
380 CALL CHARSET :: CALL SCREEN(2)
390 FOR I=1 TO 100 :: CALL HCHAR(RND*21+2,RND*30+1,91):: NEXT I
400 DATA 96,100,64,68,72,76,80,84
410 RESTORE 400
420 FOR I=1 TO 4 :: FOR K=1 TO 2 :: READ CU(I,K):: NEXT K :: NEXT I
430 CALL TITEL(COL(),B(),CPR$,STV):: P,UN=0 :: SPAR=4
440 CALL ANZSPAR(SPAR):: CALL PUNKTE(P)
450 CALL MINICHAR
460 CALL COLOR(1,2,6,3,2,6,4,2,6,8,11,1)
470 CALL SPRITE(£5,132,6,185,25,£4,104,6,185,110,£6,129,6,185,195)
480 FOR I=4 TO 6 :: FOR K=184 TO 169 STEP -1 :: CALL POSITION(£I,Z,S):: CALL LOC
ATE(£I,K,S):: CALL SOUND(-100,I*100-K,0,I*100-K+2,0,I*100-K-2,0):: NEXT K :: NEX
T I
490 CO=1
500 TRSS,STV,TRUFO=0
510 DATA 7,9,10,11,12,15,16,8,4,14
520 UN=UN+1 :: IF UN>4 THEN UN=1
530 FOR I=4 TO 6 :: CALL MOTION(£I,0,0):: NEXT I :: XA=0
540 CALL SPRITE(£1,108,11,89,30,0,0)
550 CALL SPRITE(£2,44,15,RND*120+20,RND*170+50):: CALL COLOR(£2,16):: FOR K=1 TO
 5 :: CALL SOUND(-20,-7,10-2*K):: NEXT K :: CALL COLOR(£2,15)
560 CALL PATTERN(£2,CU(UN,1)):: CALL COLOR(£2,COL(CO))
570 CALL COLOR(£4,3)
580 CALL POSITION(£2,VZ2,VS2):: IF VZ2<60 THEN SG=1 ELSE IF VZ2>150 THEN SG=-1 E
LSE SG=SGN(RND*1.5-1)
590 CALL MOTION(£2,RND*2*SG,RND*10*(SGN(RND*1.5-1))):: CALL COLOR(8,COL(RND*9+1)
,1)
600 FOR LF=1 TO 10
610 CO=CO+1 :: IF CO>10 THEN CO=1
620 CALL COLOR(£2,COL(CO))
630 FOR I=1 TO 2 :: CALL JOYST(1,X,Y):: CALL PATTERN(£4,C(I),£2,CU(UN,I))
640 CALL POSITION(£1,VZ1,VS1,£2,VZ2,VS2,£4,VZ3,VS3):: IF ABS(VS2-VS3)<12 THEN 76
0
650 IF X=XA THEN 670
660 FOR K=4 TO 6 :: CALL MOTION(£K,0,-X*5):: NEXT K
670 IF X=0 THEN 690
680 IF X=-4 THEN CALL PATTERN(£1,116)ELSE CALL PATTERN(£1,108)
690 CALL MOTION(£1,-Y*3,X)
700 CALL COINC(£1,£2,12+RUNDE,ST):: IF ST THEN 1280
710 CALL KEY(1,T,S):: IF T=18 THEN 1140
720 XA=X
730 NEXT I
740 NEXT LF
750 GOTO 580
760 CALL SPRITE(£3,60,16,VZ2+8,VS2,127,0):: CALL SOUND(-50,-3,0):: CALL POSITION
(£4,VZ3,VS3):: CALL DELSPRITE(£3):: IF ABS(VS2-VS3)<15 THEN Q=4 :: GOSUB 780 ::
GOTO 840
770 GOTO 650
780 CALL MOTION(£1,0,0)
790 CALL PATTERN(£Q,36):: CALL COLOR(£Q,12):: CALL SOUND(-100,-7,0)
800 CALL PATTERN(£Q,40):: CALL SOUND(-100,-6,0)
810 CALL PATTERN(£Q,44):: CALL SOUND(-100,-5,0)
820 CALL COLOR(£Q,16)
830 RETURN
840 CALL POSITION(£4,VZ3,VS3)
850 CALL DELSPRITE(£4)
860 CALL MOTION(£1,0,0,£2,0,0,£4,0,0,£5,0,0,£6,0,0):: XA=0
870 TRUFO=TRUFO+1 :: STV=-1 :: IF TRUFO=5 THEN 950
```




```
880 CALL SPRITE(£4,104,6,185,VS3)
890 TRO=200+100*TRUFO
900 FOR K=184 TO 169 STEP -1 :: CALL POSITION(£4,Z,S):: CALL LOCATE(£4,K,S):: CA
LL SOUND(-100,TRO-K,0,TRO-K+2,0,TRO-K-2,0):: NEXT K
910 GOTO 520
920 RANDOMIZE ASC(SEG$(CPR$,9,1))
930 FOR I=8 TO 20 :: SUM=SUM+INT(RND*ASC(SEG$(CPR$,I,1))):: NEXT I
940 GOTO 1460
950 CALL DELSPRITE(ALL):: CALL SPRITE(£2,120,14,247,120,2,0):: CALL SIRENE
960 CALL MOTION(£2,0,0)
970 C1=1 :: CALL COLOR(£2,COL(C1))
980 CALL SPRITE(£1,108,11,89,30,0,0)
990 CALL LOCATE(£2,89,120)
1000 CALL POSITION(£2,VZ2,VS2):: IF VZ2<60 THEN SG=1 ELSE IF VZ2>150 THEN SG=-1
ELSE SG=SGN(RND*1.5-1)
1010 CALL MOTION(£2,RND*2*SG,RND*10*(SGN(RND*1.5-1))):: CALL COLOR(8,COL(RND*9+1
),1)
1020 FOR LF=1 TO 11-RUNDE
1030 CO=CO+1 :: IF CO>10 THEN CO=1
1040 FOR I=3 TO 4
1050 CALL JOYST(1,X,Y):: CALL POSITION(£1,VZ1,VS1,£2,VZ2,VS2):: CALL PATTERN(£2,
C(I))
1060 IF X=0 THEN 1080
1070 IF X=-4 THEN CALL PATTERN(£1,116)ELSE CALL PATTERN(£1,108)
1080 CALL MOTION(£1,-Y*3,X*3)
1090 CALL COINC(£1,£2,12+RUNDE,ST):: IF ST THEN 1430
1100 CALL KEY(1,T,S):: IF T=18 THEN 1310
1110 NEXT I
1120 NEXT LF
1130 GOTO 1490
1140 IF X=0 THEN 720
1150 CALL MOTION(£1,0,0)
1160 CALL POSITION(£1,VZ1,VS1,£2,VZ2,VS2)
1170 IF (VS1+2*X)<1 OR(VS1+2*X)>255 THEN 720
1180 CALL SPRITE(£3,92,16,VZ1,VS1+2*X,0,31*X):: CALL SOUND(-50,-2,0):: IF SGN(VS
2-VS1)<>SGN(X)THEN CALL DELSPRITE(£3):: GOTO 720
1190 CALL DELSPRITE(£3):: IF ABS(VZ1-VZ2)<12 THEN Q=2 :: GOSUB 780 :: GOTO 1210
1200 GOTO 720
1210 P=P+10*CO*UN :: CALL PUNKTE(P):: TRSS=TRSS+1 :: IF TRSS=10 THEN 1230
1220 GOTO 520
1230 TRSS=0 :: RUNDE=RUNDE+1 :: IF STV THEN 1260 ELSE SPAR=SPAR+1
1240 IF SPAR>15 THEN SPAR=15
1250 CALL ANZSPAR(SPAR):: CALL BIM
1260 IF RUNDE>8 THEN RUNDE=8
1270 CALL NEXTR :: STV=0 :: GOTO 520
1280 CALL DELSPRITE(£2):: Q=1 :: GOSUB 780
1290 CALL DELSPRITE(£1):: CALL COLOR(£1,1):: SPAR=SPAR-1 :: CALL ANZSPAR(SPAR)::
 IF SPAR<0 THEN 1480
1300 GOTO 520
1310 IF X=0 THEN 1110
1320 CALL MOTION(£1,0,0)
1330 CALL POSITION(£1,VZ1,VS1,£2,VZ2,VS2)
1340 IF (VS1+2*X)<1 OR(VS1+2*X)>255 THEN 1110
1350 CALL SPRITE(£3,92,16,VZ1,VS1+2*X,0,31*X):: CALL SOUND(-50,-2,0):: IF SGN(VS
2-VS1)<>SGN(X)THEN CALL DELSPRITE(£3):: GOTO 1110
1360 CALL DELSPRITE(£3):: IF ABS(VZ1-VZ2)<12 THEN 1380
1370 GOTO 1110
1380 P=P+50*C1 :: CALL PUNKTE(P):: C1=C1+1 :: IF C1>10 THEN 1410
1390 CALL SOUND(-100,-7,0):: CALL SOUND(-100,-6,0):: CALL SOUND(-100,-5,0)
1400 CALL COLOR(£2,COL(C1)):: GOTO 1110
1410 Q=2 :: GOSUB 780 :: CALL DELSPRITE(ALL):: RUNDE=RUNDE+1 :: IF RUNDE>8 THEN
RUNDE=8
```

```
1420 GOTO 470
1430 CALL DELSPRITE(£3):: Q=1 :: GOSUB 780
1440 CALL COLOR(£1,1):: SPAR=SPAR-1 :: CALL ANZSPAR(SPAR):: IF SPAR<0 THEN 1480
1450 GOTO 980
1460 SKP=INT(RND*VAL(SEG$(CPR$,3,4)))-184-2*SUM
1470 IF SKP THEN 1600 ELSE 270
1480 CALL DELSPRITE(ALL):: STV=-1 :: CALL CHARSET :: GOTO 430
1490 CALL MOTION(£1,0,0,£2,0,0)
1500 CALL POSITION(£1,VZ1,VS1,£2,VZ2,VS2):: CALL SPRITE(£3,136,10,VZ2,VS2,(VZ1-V
Z2)/2,(VS1-VS2)/2)
1510 FOR I=1 TO 2
1520 FOR K=1 TO 2
1530 CALL SOUND(-100,-7,I*K)
1540 CALL PATTERN(£2,C(K+2),£3,C(K+4))
1550 CALL JOYST(1,X,Y):: CALL MOTION(£1,-Y*6,X*6)
1560 CALL COINC(£1,£2,11+R,ST1):: CALL COINC(£1,£3,18,ST2):: IF ST1 OR ST2 THEN
1430
1570 NEXT K
1580 NEXT I
1590 CALL DELSPRITE(£3):: CALL MOTION(£1,0,0):: GOTO 1000
1600 ON BREAK NEXT
1610 CALL SCREEN(16):: CALL CLEAR
1620 CALL SOUND(-1000,1000,0,998,0,1002,0):: GOTO 1620
1630 SUB BIM
1640 CALL MOTION(£1,0,0,£4,0,0,£5,0,0,£6,0,0):: CALL DELSPRITE(£2)
1650 FOR I=0 TO 30 :: CALL SOUND(-100,523,I,525,I,521,I):: NEXT I
1660 SUBEND
1670 SUB NEXTR
1680 CALL MOTION(£1,0,0,£4,0,0,£5,0,0,£6,0,0):: CALL DELSPRITE(£2)
1690 FOR I=30 TO 0 STEP -2
1700 CALL SOUND(-100,-7,I):: CALL SOUND(-100,-6,I):: CALL SOUND(-100,-5,I):: CAL
L SOUND(-100,-6,I):: NEXT I
1710 SUBEND
1720 SUB MINICHAR
1730 RESTORE 2240
1740 FOR I=32 TO 95 :: READ C$ :: CALL CHAR(I,C$):: NEXT I
1750 SUBEND
1760 SUB ANZSPAR(SPAR)
1770 CALL HCHAR(1,3,32,15)
1780 IF SPAR<=0 THEN SUBEXIT
1790 CALL HCHAR(1,3,59,SPAR)
1800 SUBEND
1810 SUB PUNKTE(P)
1820 DISPLAY AT(1,23)SIZE(6):USING "££££££":P
1830 SUBEND
1840 SUB TITEL(COL(),B(),CPR$,STV)
1850 DATA S,I,L,V,E,R,S,P,A,R
1860 DATA G,A,M,E,O,V,E,R
1870 IF STV THEN RESTORE 1860 ELSE RESTORE 1850
1880 CALL MAGNIFY(2)
1890 IF STV THEN 1910
1900 CALL HCHAR(24,1,32,64)
1910 CALL CHAR(91,"0000001818",64,"3C4299A1A199423C"):: FOR I=1 TO 7 :: CALL COL
OR(I,1,8):: NEXT I :: CALL COLOR(8,11,1)
1920 IF STV THEN 2030
1930 S=48 :: Z=17
1940 FOR I=1 TO 10 :: GOSUB 1990 :: READ C$ :: CALL SPRITE(£I,ASC(C$),COL(I),Z,S
):: Z=Z+16 :: S=S+16 :: NEXT I
1950 DISPLAY AT(24,5)SIZE(20):CPR$
1960 CALL FLYINGDREAM(B(),1)
1970 CALL DELSPRITE(ALL):: CALL MAGNIFY(3):: CALL HCHAR(24,1,32,64):: CALL COLOR
(3,8,8,4,8,8)
```

```
1980 GOTO 2110
1990 CALL SPRITE(£I,42,COL(I),Z,S):: CALL SOUND(-100,-7,0)
2000 CALL PATTERN(£I,43):: CALL SOUND(-100,-6,0)
2010 CALL PATTERN(£I,91):: CALL SOUND(-100,-5,0)
2020 RETURN
2030 ZP,SP=72 :: I=1
2040 FOR J=0 TO 1
2050 FOR K=0 TO 3
2060 Z=ZP+J*32 :: S=SP+K*32 :: READ C$ :: GOSUB 1990 :: CALL SPRITE(£I,ASC(C$),C
OL(I),Z,S)
2070 I=I+1
2080 NEXT K
2090 NEXT J
2100 GOTO 1950
2110 SUBEND
2120 SUB SIRENE
2130 K=0
2140 FOR I=261 TO 293 STEP 10 :: CALL SOUND(-250,I,0,I-2,0):: NEXT I :: CALL PAT
TERN(£2,124)
2150 K=K+1
2160 FOR I=303 TO 523 STEP 10 :: CALL SOUND(-250,I,0,I-2,0,I-42,3):: NEXT I :: C
ALL PATTERN(£2,120)
2170 FOR I=523 TO 491 STEP -10 :: CALL SOUND(-250,I,0,I-2,0,481-(523-I),3):: NEX
T I :: CALL PATTERN(£2,124)
2180 FOR I=481 TO 261 STEP -10 :: CALL SOUND(-250,I,0,I-2,0,I+42,3):: NEXT I ::
CALL PATTERN(£2,120)
2190 IF K=3 THEN 2220
2200 FOR I=261 TO 293 STEP 10 :: CALL SOUND(-250,I,0,I-2,0,293+(293-I),3):: NEXT
 I :: CALL PATTERN(£2,124)
2210 GOTO 2150
2220 SUBEND
2230 SUB GRAPH
2240 DATA ,,,,00000001E273351F,0B060F1B3E3F61C1,2870E0C08CD8F058
2250 DATA CCE6B3E0C08,00001800C233141F,0B000F110E3B01C1,287086400C18F058,CCE2B3E
0028C,1000100080220411,0A000510042A0041
2260 DATA 2000044004100005,048022800288,0018242424242418,0010301010101038,00182404081
03C,00182408042418,000818283C0808
2270 DATA 003C2038042418,00182038242418,003C040810101,00182418242418,0018242410C0
418,,0070380E0E387
2280 DATA 0101010101010101,0101010101010101,8080808080808080,8080808080808080,0103
070F1E3F7FE3,E37F3F1F0C0C0C3C,80C0E0F078FCFEC7
2290 DATA C7FEFCF83030303C,00000000183C7EE7,E37F3F1F0C0E,00000000183C7EE7,C7FEFC
F8307,20100805030F3FFF,C1FF090D0701
2300 DATA 040810A0C0F0FCFF,83FF90B0E08,0000003C030F3FFF,F3FF09080818,0000003CC0F
0FCFF,CFFF90101018,003C66C3FF7F3F01
2310 DATA 03060C060301,003C66C3FFFEFC8,C0603060C08,00007EC3FF1C3F01,030606060301
,00007EC3FF38FC8,C0606060C08
2320 DATA ,,,0000001818,00000000000000FF,FF,00000000000000FF,FF
2330 DATA 000000010307077F,FFCEFF7F,00000080C0E0E0FE,FF73FFFE,0000000000003077F,F
FF1FF7F
2340 DATA 00000000000C0E0FE,FF8FFFFE,000103070D0B0F07,03020303070D1F3B,0080C060B0
B070E,C0C040C0E070B8DC,00307FD8A05F2C17
2350 DATA 172C5FA0D87F3,0000800000F00CFF,FF0CF000008,000103060D0D0E07,0303020307
0E1D3B,0080C0E0B0D0F0E,C040C0C0E0B0F8DC
2360 DATA 00000100000F30FF,FF300F000001,000CFE1B05FA34E8,E834FA051BFE0C,83CFFF7F
61333E3D,1D1C1F0C07030301,C1F3FFFE86CC7CBC
2370 DATA B838F830E0C0C08,034FFF7F73213F3E,1D3C3F1E0F070301,C0F2FFFECE84FC7C,B83
CFC78F0E0C08,000000000000003F,7FDBDBFFFFFFFFFF
2380 DATA 04040A0A1B1F1F95,D57F7FFFFFFFFBFB,0000081C367FFF49,7F7F77777F7FFFFF,00
0000000008,0000000080C0E0F,0001030606034143
2390 DATA 673D1800000003,00C000000018BCE6,C282C06060C08,00070C18180C0701,4342663
```

```
C18,000000183C6642C2,80E030181830E
2400 RESTORE 2330
2410 FOR I=96 TO 143 :: READ C$ :: CALL CHAR(I,C$):: NEXT I
2420 SUBEND
2430 SUB FLYINGDREAM(B(),KY)
2440 DATA 1,195,261,329,391,523,659,1,261,329,1,293,349,4,329,391
2450 DATA 1,523,440,349,261,220,174,1,329,391,1,349,440,4,293,349
2460 DATA 1,195,246,349,391,493,698,1,293,349,1,261,329,2,246,293,2,293,349
2470 DATA 1,659,523,391,329,261,195,6,261,329
2480 DATA 2,30000,30000,30000,30000,30000,30000,1,30000,329,1,30000,369,2,30000,
293,2,30000,440,1,30000,440
2490 DATA 1,30000,493,4,30000,391,1,174,174,220,220,261,261,2,30000,30000,2,293,
349,2,349,440
2500 DATA 1,195,195,246,246,293,293,6,329,391,4,195,261,329,391,523,659
2510 DATA 1,195,329,1,174,293,2,164,261,2,329,391
2520 DATA 1,195,329,1,174,293,2,164,261,2,329,391
2530 DATA 1,195,329,1,174,293,2,164,261,2,329,391
2540 DATA 6,164,261,0
2550 RESTORE 2440
2560 AZ,STV=0
2570 READ K :: IF K=0 THEN 2550
2580 READ B(1),B(2),B(3),B(4),B(5),B(6)
2590 FOR I=1 TO K
2600 FOR J=1 TO 6
2610 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
2620 CALL SOUND(350,M1,1,M2,0,B(J),5):: CALL KEY(1,T,S):: IF S<>0 THEN SUBEXIT
2630 NEXT J :: NEXT I
2640 GOTO 2570
2650 SUBEND
```

TRANSFER

Das Programm TRANSFER gehört mit zu den ASCII Hilfsprogrammen und erweitert dieses Programmpaket.

Das Unterprogramm ASCIISET belegt den Zeichensatz durch das mittels des Programmes ASCIIDEF erstellte Codefeld und erfordert immer einen externen Speicher. Mittels des Programmes TRANSFER wird aber vom Rechner selbst ein Unterprogramm geschrieben, welches den Zeichensatz im Programm, also ohne externen Speicher, umbelegt.
TRANSFER ist also ein Programm, welches aus dem Codefeld ein Unterprogramm schreibt.
Dieses Unterprogramm kann dann mittels MERGE auf das gewünschte BASIC-Hauptprogramm aufgeladen werden.
Das Programm TRANSFER ist nur mittels eines Diskettenlaufwerkes sinnvoll einzusetzen, da man von einem Kassettenrecorder kein Programm mit MERGE aufladen kann.
Nach dem Programmstart wird man nach dem entspr. Codefeld gefragt, welches mit Gerät.-Name (z. B. DSK1.FELD) eingegeben werden muß. Danach wird man nach dem Namen gefragt, unter welchem das Unterprogramm auf Diskette gespeichert werden soll (z. B. DSK1.PGM).
Zuletzt kommt die Frage nach dem Namen des zu erstellenden Unterprogrammes (z. B. ASCII).
Der Rechner generiert jetzt das Unterprogramm, in folgender Form:
Das Unterprogramm belegt die Zeilen 32000 bis 32019. In der ersten Zeile steht SUB und der eingegebene Name Zeilen 32001 bis 32016 sind DATA-Zeilen, in welchen der komplette hexadezimale Zeichensatz abgelegt ist.
Zeile 32017 ist die RESTORE Anweisung.
Zeile 32018 enthält die Schleife zur Umbelegung des Zeichensatzes.
In Zeile 32019 steht SUBEND.
Das so generierte Unterprogramm kann nun mit MERGE auf ein bestehendes BASIC-Programm aufgeladen werden und wird im Programm mit CALL Name (z. B. CALL ASCII) aufgerufen und belegt den kompletten Zeichensatz um, wie er mittels „ASCIIDEF" generiert wurde.
Der TI-99/4 A programmiert sich somit praktisch selbst.

```
100 REM ******************************
110 REM *                            *
120 REM *       T R A N S F E R      *
130 REM *----------------------------*
140 REM *    TI 99/4 A               *
150 REM * + TI Extended BASIC        *
160 REM * + Diskettenlaufwerk        *
170 REM *----------------------------*
180 REM * (c) 1984 by Volker Becker  *
190 REM *                            *
200 REM ******************************
```



```
210 ON WARNING NEXT
220 DISPLAY AT(1,1)ERASE ALL:"Erstellen eines BASIC Unter- Programmes zur Umbele
gung           des Zeichensatzes        ----------------------------"
230 ON ERROR 710
240 DISPLAY AT(6,1):"Name der Eingabedatei          (Codefeld) :" :: ACCEPT AT(7,14
)BEEP:IDAT$
250 OPEN £1:IDAT$,INPUT ,SEQUENTIAL,INTERNAL,FIXED 80
260 ON ERROR 720
270 DISPLAY AT(9,1):"Ausgabedatei (Name unter     welchem das Programm auf     Dis
kette gespeichert werden soll)"
280 ACCEPT AT(12,7)BEEP:ODAT$
290 OPEN £2:ODAT$,OUTPUT,SEQUENTIAL,DISPLAY ,VARIABLE 163
300 ON ERROR STOP
310 DISPLAY AT(14,1):"Unterprogrammname :" :: ACCEPT AT(15,1)BEEP:NAME$
320 DISPLAY AT(18,1):"Unterprogramm ";NAME$ :: DISPLAY AT(19,1):"wird erstellt.
Bitte warten"
330 PRINT £2:CHR$(125)&CHR$(0)&CHR$(161)&CHR$(200)&CHR$(LEN(NAME$))&NAME$&CHR$(0
)
340 FOR I=1 TO 16
350 A$=CHR$(125)&CHR$(I)&CHR$(147)
360 FOR K=1 TO 7
370 INPUT £1:G$,HEX$
380 FOR J=16 TO 1 STEP -1
390 IF SEG$(HEX$,J,1)<>"0" THEN 420
400 NEXT J
410 GOTO 480
420 HEX$=SEG$(HEX$,1,J)
430 A$=A$&CHR$(200)&CHR$(LEN(HEX$))
440 FOR J=1 TO LEN(HEX$)
450 A$=A$&SEG$(HEX$,J,1)
460 NEXT J
470 IF K=7 THEN 490
480 A$=A$&CHR$(179)
490 NEXT K
500 A$=A$&CHR$(0)
510 PRINT £2:A$
520 NEXT I
530 DATA 125,17,148,201,125,1,0
540 DATA 125,18,140,73,190,200,2,51,50,177,200,3,49,52,51,130,151,67,36,130,157,
200,4,67,72,65,82
550 DATA 183,73,179,67,36,182,130,150,73,0
560 DATA 125,19,168,0
570 RESTORE 530
580 FOR I=1 TO 3
590 A$=""
600 READ WERT
610 A$=A$&CHR$(WERT)
620 IF WERT=0 THEN 630 ELSE 600
630 PRINT £2:A$
640 NEXT I
650 PRINT £2:CHR$(255)&CHR$(255)
660 CLOSE £1 :: CLOSE £2
670 DISPLAY AT(18,1):"Unterprogramm ";NAME$ :: DISPLAY AT(19,1):"ist erstellt un
d kann mit    MERGE ";ODAT$
680 DISPLAY AT(21,1)BEEP:"auf ein BASIC Programm       aufgeladen werden."
690 STOP
700 END
710 CLOSE £1 :: GOTO 240
720 CLOSE £2 :: GOTO 280
```



Dateiprogramme für den Commodore 64 inkl. Diskettenlaufwerk VC 1541.

Dateiprogramme gibt es ja schon eine Menge, jedoch meist nur für eine gezielte Anwendung. Das hier vorgestellte Programm gibt jedem die Möglichkeit sich sein individuelles Dateiprogramm zu erzeugen. Dabei spielt es keine Rolle, ob man Videofilme, Adressen oder etwas anderes verwalten möchte. Jeder Datensatz kann außer dem Datum der Eintragung noch bis zu 15 weitere Eintragungen enthalten. Die unterschiedlichen Kriterien werden einfach in die DATA-Zeile 60020 eingetragen. Wichtig ist, daß diese Zeile mit einem Komma abgeschlossen wird.

Z. B. Adressverwaltung:
60020 DATA Vorname, Nachname, Straße, Ort, Telefon, Geburtstag, Bemerkung.

Z. B. Videofilme:
60030 DATA Titel, Länge, Regisseur, Musik, Hauptdarsteller.

Vor dem ersten Start muß in Zeile 80 der zweite GOSUB-Befehl entfernt werden, da sonst versucht wird, eine nicht vorhandene Datei einzuladen. Nach dem Starten stehen folgende Befehle zur Verfügung:
(Z) Summieren
(O) Sortieren
(S) Suchen
(A) Daten ausgeben
(N) Neuer Antrag
(Q) Daten saven
Der Befehl „Z" summiert einen beliebigen Eintrag.
Der Befehl „O" sortiert alphabetisch nach einem bestimmten Eintrag.
Der Befehl „S" sucht ein bestimmtes Kriterium durch.
Der Befehl „A" gibt einzelne; mehrere oder auch alle Daten auf Drucker aus. Ebenfalls ist es auch möglich Adressaufkleber zu drucken.
Der Befehl „N" erlaubt einen neuen Eintrag.
Der Befehl „Q" speichert die Daten auf Diskette.
Als weitere Unterbefehle gibt es Edit zum editieren und Clear zum löschen von Einträgen.
Je nach Anzahl der Kriterien dürfte die durchschnittliche Anzahl der Datensätze bei 400 liegen. Da jedoch mit mehreren Files gearbeitet werden kann, ist diese Zahl beliebig zu vervielfachen.
Das Programm arbeitet weitgehend mit Masken und Menüs, so daß eine Fehlbedienung ausgeschlossen ist.

```
10 REM *****************
20 REM    C64 DATENBANK
30 REM *****************
50 :
60 DIM S$(16),DD$(500,16),DF$(16)
66 :
69 POKE 53281,14:POKE 53280,5:U=1
70 READ S$(U):IF S$(U)<>"" THEN U=U+1:GOTO 70
75 S$(0)="DATUM"
80 GOSUB 53000:    GOSUB 40030
100 PRINT"■■ DATENBANK                    "
110 FORI=0 TO U-1
120 PRINT"<"I">  "TAB(9)S$(I)
130 NEXTI
135 H%=28:V%=1:GOSUB 50030:PRINTDA$
136 H%=28:V%=2:GOSUB 50030:PRINT LEFT$(FI$,10)
140 H%=0:V%=18:GOSUB 50030
145 PRINT"<Z> SUMMIEREN         <O> SORTIEREN"
150 PRINT"<S> SUCHEN            <A> DATEN AUSGEBEN"
160 PRINT"<N> NEUER EINTRAG    <Q> DATEN SAVEN"
190 PRINT:PRINT"BEFEHL -->              DATENANZ.:"O
200 GET E$:IF E$=""THEN 200
210 IF E$="N" THEN 1000
220 IF E$="Q" THEN 40140
230 IF E$="S" THEN 2000
240 IF E$="Z" THEN 3000
245 IF E$="O" THEN 4000
247 IF E$="A" THEN 6000
250 GOTO 200
1000 REM *** NEUER EINTRAG ***
1010 FI=O:DD$(FI,0)=DA$:GOSUB20000
1015 FI=O:GOSUB 30000
1017 PRINT:PRINT:PRINT" ALLES IN ORDNUNG (J/N) ?"
1018 GET E$:IF E$=""THEN 1018
1019 IF E$="N"THEN 1010
1020 O=O+1
1030 GOTO 100
2000 :
2010 REM *** SUCHEN ***
2020 :
2030 H%=0:V%=22:GOSUB 50030:PRINT"■SUCHEN■ NACH WELCHEM
     KRITERIUM",
2040 INPUT E$
2050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 2030
2060 H%=0:V%=22:GOSUB 50030:FORJ=0 TO 38:PRINT" ";:NEXTJ
2070 H%=0:V%=22:GOSUB 50030:I=VAL(E$):PRINT S$(I)" ";:M=I
2080 INPUT SW$
2090 GOSUB 51000
2095 IF FI<0 THEN 100
2100 GOSUB 20000
2110 H%=0:V%=19:GOSUB 50030:PRINT"■ <N> NEXT <E> EDIT <M> MENUE <C> CLEAR  ■"
2120 GET E$:IF E$="" THEN 2120
2130 IF E$="M" THEN 100
2140 IF E$="N" THEN I=M:T=T+1:GOSUB 51040:GOTO 2095
2150 IF E$="C" THEN 52000
2160 IF E$="E" THEN GOSUB 30000:DD$(FI,0)=DA$:GOTO 2095
2170 GOTO 2120
3000 :
3005 REM *** EINTRAG SUMMIEREN ***
3010 :
3030 H%=0:V%=22:GOSUB 50030:PRINT"■ SUMME ■ VON WELCHEM KRITERIUM  ";
3040 INPUT E$
3050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 3030
```

```
3060 H%=0:V%=22:GOSUB 50030:FORJ=0 TO 38:PRINT" ";:NEXTJ
3061 S2=0:I=VAL(E$)
3062 FORT= 0 TO O-1:S1=VAL(DD$(T,I))
3063 S2=S2+S1:NEXTT
3070 H%=0:V%=22:GOSUB 50030:PRINT"SUMME = "S2
3080 GET E$:IFE$=""THEN3080
3090 GOTO100
4000 :
4010 REM *** SORTIEREN ****
4020 :
4030 H%=0:V%=22:GOSUB 50030:PRINT"▩SORTIEREN▩ NACH WELCHEM KRITERIUM  ";
4040 INPUT E$
4050 IF VAL(E$)<0 OR VAL (E$)>U-1 THEN 4030
4060 T=VAL(E$)
4070 FOR L1=0 TO O-2:FOR L2=L1+1 TO O-1
4080 GOSUB 5000
4090 IF Q1<Q2 THEN 4110
4100 GOSUB 4130
4110 NEXT:NEXT:GOTO 100
4120 REM UNTERPROGRAMME FUER BUBBLESORT
4130 FOR I=0 TO U-1
4140 DP$(I)=DD$(L1,I)
4150 DD$(L1,I)=DD$(L2,I)
4160 DD$(L2,I)=DP$(I)
4170 NEXTI:RETURN
5000 Z=1
5010 Q1$=MID$(DD$(L1,T),Z,1)
5020 Q2$=MID$(DD$(L2,T),Z,1)
5030 IF Z>15 THEN 5050
5040 IF Q1$=Q2$ THEN Z=Z+1:GOTO 5010
5050 Q1=0:IF Q1$<>"" THEN Q1=ASC(Q1$)
5060 Q2=0:IF Q2$<>"" THEN Q2=ASC(Q2$)
5070 RETURN
6000 :
6010 REM *** DATEN AUSGEBEN ***
6020 :
6030 PRINT"❏ *** DATEN AUSGEBEN ***▩"
6040 FORI=0 TO U-1
6050 PRINT"<"I">  "TAB(9)S$(I)
6060 NEXTI
6070 PRINT"< A >     ALLES   AUSGEBEN"
6080 PRINT"< E >     4 DATEN AUSGEBEN  Z.B. ADRESSE"
6090 PRINT:PRINT"BEFEHL -->";
6095 E$="":FORI=1 TO 100:NEXTI
7000 GET E$:IF E$=""THEN7000
7010 IF E$="A"THEN 8000
7020 IF E$="E"THEN 9000
7030 I=VAL(E$):
7035 OPEN 4,4
7036 PRINT#4,"LISTE VOM "DA$
7037 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
7040 FORT=0 TO O-1
7050 PRINT#4,T;CHR$(16)"05"DD$(T,I)
7060 NEXTT
7080 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
7085 CLOSE4
7090 GOTO 100
8000 :
8010 REM *** ALLES AUSGEBEN ***
```

Tips & Tricks

Reset für die kleinen Commodore!

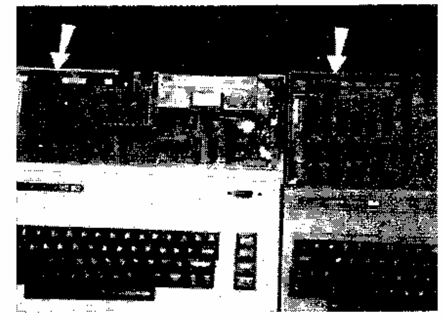
Wer kennt dies nicht?

Ich schreibe ein Programm (maschinenspr.), gebe ein paar Pok's zur Probe ein. Was ist nun los? ... Cursor futsch ... ... nichts geht mehr ...
Eine einzige Möglichkeit bleibt offen, das ist der berühmte große Knopf. – Hauptschalter aus, Hauptschalter ein, nun scheint alles wieder okay zu sein! Rein optisch laut Bildschirm ist auch alles wieder in Ordnung, aber das Programm ist und bleibt futsch!!
Ein selbstgepinntes Programm. Stunden sitze ich nun schon daran, abgespeichert war es nicht, schließlich war das Ende des Listings doch so nahe. Dann kann ich ja das ganze Programm absaven, dacht ich mir.
Aber Prosit – Mahlzeit, ich halte die Luft an, starre auf den Bildschirm voller Entsetzen, Cursor weg – nichts geht mehr, mir wirds heiß, ich spüre wie mir das Blut in den Kopf steigt, meine Gehirnzellen arbeiten plötzlich fieberhaft. Hilfe, mein Programm ist futsch! Jetzt wird mir erst bewußt, das die ganze „Tipperitis" für die Katz war. Oh', wie bin ich von meinem Computer enttäuscht. Dieser Computer, der mein bester Freund ist, entpuppt sich zum Mistkerl Nr. 1. Wie kann er mir so etwas antun?? Das schlimme an dieser Sache ist, dies war nicht das erstemal, er hat sich diese Mätzchen schon ein oder zweimal erlaubt.

Nun ist aber Schluß mit den Amokläufern

Sofort baue ich ihm eine Reset-Taste ein. Man braucht kein Elektroniker oder gar ein Computer-Mechaniker zu sein, um dies zu bewerkstelligen. Sehen wir uns hierzu die Handbücher einmal an:

VC-20 Seite 152
CBM 64 Seite 143



Der Commodore 64 und VC-20 geöffnet. – USER PORT – oben links (siehe Pfeil)

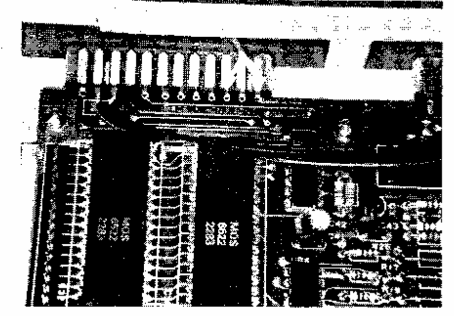
Dort finden wir den für uns wichtigen USER-PORT. Wir brauchen Pin 1

```
8020 :
8030 OPEN 4,4
8040 PRINT#4,"GESAMTLISTE VOM "DA$
8050 PRINT#4
8060 FOR FI=0 TO O-1
8070 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
8080 FORT=0 TO U-1
8082 PRINT#4,S$(T);CHR$(16)"16:  "DD$(FI,T)
8083 NEXTT
8090 NEXT FI
8095 FORT=0 TO 78:PRINT#4,"▒";:NEXTT:PRINT#4
8100 CLOSE4
8110 GOTO100
9000 :
9010 REM *** 4 DATEN AUSGEBEN ***
9020 :
9030 PRINT"▯DREI DATEN AUSGEBEN"
9040 PRINT"--------------------------------------"
9041 FORI=0 TO U-1
9042 PRINT"<"I">  "TAB(9)S$(I)
9043 NEXTI
9045 PRINT"▮BEACHTEN SIE DAS DIE ERSTEN BEIDEN"
9046 PRINT"DATEN HINTEREINANDER GEDRUCKT
     WERDEN !▮▮"
9050 INPUT "DATEN KENNZAHL 1 ";D1
9060 INPUT "DATEN KENNZAHL 2 ";D2
9070 INPUT "DATEN KENNZAHL 3 ";D3
9080 INPUT "DATEN KENNZAHL 4 ";D4
9100 OPEN 4,4
9110 FORT=0 TO O-1
9120 PRINT#4,DD$(T,D1);" "DD$(T,D2)
9130 PRINT#4,DD$(T,D3)
9140 PRINT#4,DD$(T,D4)
9150 PRINT#4:PRINT#4
9160 NEXTT
9170 CLOSE 4
9180 GOTO100
20000 REM *** DATENSATZ ANZEIGEN ***
20010 REM FI=NR DES DATENSATZES
20020 :
20030 PRINT"▯":PRINT
20035 PRINTS$(0)":  "DD$(FI,0)
20040 FORI=1 TO U-1
20050 PRINT LEFT$(S$(I),16);TAB(18)""DD$(FI,I)
20060 NEXTI:RETURN
20070 :
30000 REM *** EINGABE ROUTINE ***
30010 REM FI=NR DES DATENSATZES
30020 FORT=1 TO U-1
30030 H%=15:V%=T+1:GOSUB 50030
30040 E$="":INPUT "";E$:IF E$<>""THEN DD$(FI,T)=E$
30050 NEXTT
30070 RETURN
40000 :
40010 REM *** DATEN EINLADEN ***
40020 :
40030 F2$=F1$+",S,R":OPEN 1,8,4,F2$
40040 INPUT#1,O
40050 FORI=0 TO U-1
```

```
40060 FORT=0 TO O-1
40070 INPUT#1,DD$(T,I)
40080 NEXT T,I
40090 CLOSE1
40100 RETURN
40110 :
40120 REM *** DATEN SAVEN ***
40130 :
40140 F2$="@:"+F1$+",S,W":OPEN 1,8,4,F2$
40150 PRINT"■■■ACHTUNG !!!!!!!"
40160 PRINT"■DATEN WERDEN GESPEICHERT !"
40170 PRINT#1,O
40180 FORI=0 TO U-1
40190 FORT=0 TO O-1
40195 IF DD$(T,I)="" THEN DD$(T,I)="-"
40200 PRINT#1,DD$(T,I)
40210 NEXT T,I
40220 CLOSE1:END
40230 :
50000 REM *** CURSOR SETZEN ***
50010 REM H%=HORIZONTAL
50020 REM V%=VERTIKAL
50030 PRINT"■";
50040 FORI=0 TO V%:PRINT"■";:NEXT
50045 IF H%=0 THEN 50060
50050 FORI=0 TO H%:PRINT"■";:NEXT
50060 RETURN
51000 :
51010 REM *** STICHWORT SUCHEN ***
51020 :
51030 T=0
51040 IF DD$(T,I)=SW$ THEN FI=T:RETURN
51050 T=T+1:IF T>O-1 THEN FI=-1:RETURN
51060 GOTO 51040
51070 :
52000 REM *** DATENSATZ LOESCHEN ***
52010 :
52020 O=O-1
52030 FOR T=FI TO O-1
52040 FOR I=0 TO U-1
52050 DD$(T,I)=DD$(T+1,I)
52060 NEXT I,T
52065 FOR T=0 TO U-1:DD$(O,T)="":NEXTT
52070 GOTO 100
52080 :
53000 REM *** FILENAMEN EINGEBEN ***
53010 :
53020 PRINT"■■ ■              D A T E N B A N K  "
53040 PRINT"■■■■":INPUT"  FILENAMEN ";F1$
53050 PRINT"■":  INPUT"  DATUM       ";DA$
53060 RETURN
60000 :
60005 REM * IN FOLGENDE DATA-ZEILE *
60006 REM * GEWUENSCHTE ARGUMENTE   *
60007 REM * EINTRAGEN. Z.B. NAME...*
60010 :
60020 DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,

READY.
```

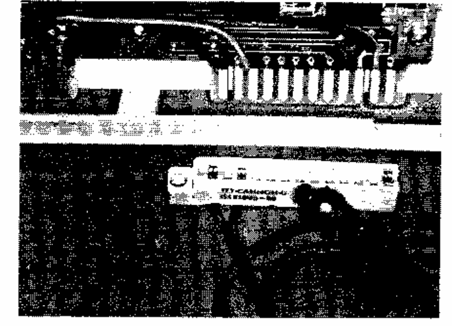
(Ground) und Pin 3 (Reset). Wenn wir nun die beiden Pin's kurz zusammen bringen, erreichen wir hierdurch einen „Kaltstart". Da wir den Strom



An Pin 1 + 3 (Pfeile) soll gelötet werden.

(Ein-Ausschalter) nicht abgeschaltet haben, bleibt der Speicherinhalt erhalten. Ein Basic-Programm scheint zwar verloren gegangen zu sein, kann aber durch VERBIEGEN der entsprechenden Zeiger wieder gerettet werden.

Es gibt nun zwei Möglichkeiten für den VC-20 genau wie für den 64er! RESET-TASTE innen oder außen. Nehmen wir einmal an, Sie sind außerhalb der Garantiezeit und öffnen Ihr Gerät, so sehen Sie den User-Port oben links, der Computer steht also mit der Front zu Ihnen hin gerichtet. Sie zählen die Pin's von rechts nach links, also ganz rechts ist Pin 1, links daneben Pin 2 usw. Auf der Platine sehen wir vor jedem Pin eine kleine Bohrung, dort löten wir einen dünnen isolierten etwa 50 cm langen Kabel an (Telefondraht). Das gleiche machen wir an der Bohrung vor Pin 3. Die beiden freien Drahtenden löten wir jetzt an den Taster.



Stecker mit Kabel und Taster

Wenn wir uns jetzt einmal das Oberteil (Tastatur) von innen betrachten, stellen wir fest, daß wir hier viel Platz und mehrere Möglichkeiten haben, um den Taster zu befestigen. Mein Vorschlag: links neben der Power-Kontroll-Leuchte, aber dies bleibt Ihnen natürlich überlassen.

Sie sagen: „Ich habe noch Garantie auf meinen Rechner!" Macht knapp die Hälfte, ist meine Antwort. Sie besorgen sich einen Stecker für den User-Port und löten die beiden Drähtchen an den Stecker und den Taster. So brauchen Sie Ihr Gerät überhaupt nicht zu öffnen.

Topprogramm

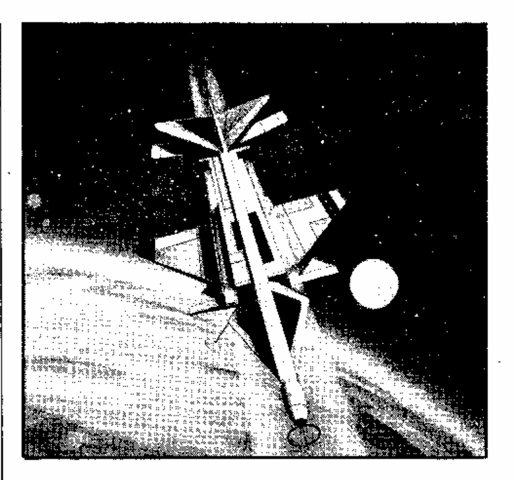
Als Topprogramm auserwählt von der Redaktion.

Sie sollen eine Reise zum Mond unternehmen, die in fünf Phasen gegliedert ist. In der ersten Phase müssen Sie Ihre Rakete unter nicht ganz einfachen Bedingungen mit dem nötigen Treibstoff für den Start und die Mondlandung beladen. Die zweite Phase dient zur eigentlichen Mondlandung, die vierte zum Start vom Mond und der Rückkehr zur Raumbasis. Dazwischen soll man möglichst viele Uranvorkommen mit Hilfe eines Geigerzählers entdecken, was Phase drei darstellt. Den Abschluß bildet die Notlandung mit Hilfe eines Fallschirmes auf der Erde (Phase 5).
Nach dem Starten des Programms fragt der Computer, falls das Spiel nach dem Einschalten zum ersten Mal läuft, ob Sie den derzeitigen Highscore eingeben wollen. Wird diese Frage durch Drücken der J-Taste beantwortet, so müssen nacheinander die Ergebnisse der einzelnen Teile eingetippt werden. Teil 6 und Teil 7 stellen dabei den Bonus für den restlichen Treibstoff (Rest Fuel) und dem Schwierigkeitsgrad (Stufe) dar, deren Werte in der Auswertung nach jedem Spiel angezeigt oder auf Wunsch ausgedruckt werden können.
Nun können Sie durch Drücken der J- bzw. N-Taste wählen, ob Sie vor jedem Teilspiel die Erklärungen lesen wollen. Entscheiden Sie sich für N, so läuft das Spiel ohne Unterbrechungen. Schließlich muß der Schwierigkeitsgrad eingestellt werden. Dies geschieht mit Hilfe des Joysticks. Ist die betreffende Zahl gewählt, den Feuerknopf drücken (Port 2).

Phase 1:
Das Ziel des ersten Spiels besteht darin, den Stapel Tanks mit Hilfe eines Männchens von der linken Seite auf die rechte zur Rakete zu transportieren. Damit dies nicht zu einfach ist, sind Minen als Hindernisse auf der Strecke, die sich nach jedem Aufladen verändern und durch Überspringen überwunden werden können.
Das Männchen wird durch den Joystick (Control Port 2) gelenkt.
Durch Drücken des Joysticks nach vorne springt es, durch Betätigen des Feuerknopfs kann ein Tank auf- bzw. abgeladen werden, wobei die Figur unbeladen weiß und beladen blau ist.
Der Spielablauf sieht also so aus: Sie laden einen Tank auf (Knopf drücken), steuern nach rechts und springen bei Bedarf über die Minen (Knüppel nach vorne), bis Sie das rechte weiße Feld erreicht haben, wo Sie den Tank abladen können (Knopf drücken). Schließlich laufen Sie zurück (auf die Minen achten!) und holen den nächsten Tank. Dieser Vorgang wiederholt sich so oft, bis die Zeit (links oben) gefährlich nahe an 10 Sek. ist, da bis dahin das Betanken abgeschlossen sein muß. Wollen Sie also starten, gehen Sie auf das rechte weiße Feld und ziehen den Steuerknüppel nach hinten, wodurch die Figur mit dem Aufzug in die Rakete befördert wird. Ist die Zeit abgelaufen, hebt die Rakete ab und der Treibstoffvorrat (Fuel) verringert sich kontinuierlich. Betrug er vor dem Start weniger als ca. 12 000 Liter, wird der Startvorgang abgebrochen und die Rakete sinkt zum Boden zurück, wo sie explodiert. Ansonsten beginnt Phase 2.
Soll das Spiel nach einem Abbruch (Crash oder ä.) neu gestartet werden, genügt das Drücken irgendeiner Taste!

Phase 2:
Wurde am Anfang nicht N eingegeben, so erscheint nun die Erklärung für Teil 2. Die Aufgabe besteht darin, eine Mondfähre auf dem Mond an einer markierten Stelle zu landen. Dabei darf man weder mit der Raumbasis noch dessen Schutzschirm kollidieren. Außerdem sind der Seitenwind, der außerhalb des Schutzschirms wirkt, und die Landegeschwindigkeit, die unten gezeigt wird, zu beachten. Sie darf nicht über einem bestimmten Wert liegen, der vom Schwierigkeitsgrad abhängt und sich folgendermaßen errechnet: LG = 0,1 x (10-GR) + 0,1 (GR = Schwierigkeit), d. h. bei Schwierigkeit 10 darf die Landegeschwindigkeit höchstens 0,1 betragen.
Die Landefähre wird durch den Joystick nach links bzw. rechts gesteuert, der Feuerknopf löst den Antrieb aus. Nach unten fällt die Fähre durch die Schwerkraft. Doch seien Sie vorsichtig, sie reagiert sehr träge! Um von der Plattform auf der Basis starten zu können, muß natürlich erst etwas Schub gegeben werden. Aber nicht zu viel, da sonst der Schutzschirm berührt wird.
Ist der Schwierigkeitsgrad kleiner als 5, so ist es möglich, nach einer Kollision oder zu harten Landung es noch einmal zu probieren. Andernfalls muß bei einer Kollision durch Drücken irgendeiner Taste von vorne begonnen werden.

Phase 3:
Dieser Teil des Spiels findet auf dem Mond statt. Mit einem bestimmten Sauerstoffvorrat sollen möglichst viele Uranvorkommen gefunden werden. Als Hilfsmittel dient ein Geigerzähler, der sich optisch durch einen Zeiger (rechts im grünen Feld) und akustisch durch ein Knacken bemerkbar macht. Mit Hilfe des Männchens muß man nun die Oberfläche so lange abschreiten, bis der Zeiger ausschlägt. Hat man die Stelle gefunden, wird sie durch Drücken des Feuerknopfes markiert, man erhält 20 Lufteinheiten als Bonus und kann das nächste Vorkommen suchen. Bevor die Lufteinheiten gleich Null sind, muß man aber wieder an der Mondfähre sein (zwischen die Pfeile stellen), um das Spiel zu beenden (Knopf drücken).
Die Figur läßt sich wieder mit dem Joystick steuern, wobei man aber nur schräg nach oben bzw. unten und nach links und rechts gehen kann. Außerdem sollten Sie Krater meiden, da ein Hineinfallen doppelten Sauerstoffverbrauch bewirkt.



Phase 4:
Ist Phase 3 beendet, erscheint das Spielergebnis, und durch Drücken des Feuerknopfes erscheint die Anleitung für Phase 4 bzw. beginnt sie.
Die Aufgabe besteht darin, zur Raumbasis zurückzukehren. Die Steuerung erfolgt wie in Phase 2. Beim Landen auf der Plattform muß keine Landegeschwindigkeit eingehalten werden.

Phase 5:
Durch Drücken des Feuerknopfes und irgendeiner Taste beginnt die letzte Phase. Da der Hauptschirm für die Kapsel versagt, muß man aussteigen und mit dem eigenen Fallschirm landen.
Wenn die Figur oben am Bildschirmrand erscheint, kann man, solange der Fallschirm noch nicht geöffnet ist, nach links bzw. rechts (Joystick)

steuern, um den von rechts kommenden Seitenwind auszugleichen, da möglichst genau auf dem weißen Feld gelandet werden soll. Will man den Fallschirm auslösen, so ist der Feuerknopf zu drücken! Geschieht dies zu spät, so zerschellt man am Boden.

Zum Abschluß erscheint die Auswertung mit den Ergebnissen der einzelnen Teile, wobei „Rest Fuel" den Bonus für den restlichen Treibstoff und „Stufe" den Bonus für den Schwierigkeitsgrad darstellen. Soll das Ergebnis ausgedruckt werden, so ist die J-Taste, ansonsten die N-Taste zu drücken.

Für ein neues Spiel ist eine Taste zu drücken.

Punktewertung: Im allgemeinen gilt für das Betanken und Uransuchen: Je später ein Tank abgeladen bzw. Uran gefunden wird, desto mehr Punkte gibt es. Damit soll die Risikobereitschaft gefördert werden. Im einzelnen folgende Formeln:

Phase 1: P = 1000/Z x 20 (Z = Zeit)
Phase 3: P = 5000/Z x 20 (Z = Lufteinheiten)
Phase 2/4: P = (1,2 - SU) x 250 (SU = Landegeschwindigkeit)
Phase 5: P = (141 - E)/0,05 (E = Entfernung vom Landepunkt)

```
1 REM *******************
2 REM * PROJEKT         *
3 REM * BY              *
4 REM * ACHIM MUELLER   *
5 REM *                 *
6 REM *                 *
7 REM * 10.05.1984      *
8 REM *******************
10 V=53248:SI=54272:DI=6144:POKEV+21,0:B=0
20 FORI=SITOSI+28:POKEI,0:NEXT
30 IFPEEK(2)=255THEN190
40 FORA=704TO720:POKEA,0:NEXT
50 PRINT"■WOLLEN SIE DEN DERZEITIGEN HIGHSCORE":PRINT"■EINGEBEN (J/N)?"
60 GETA$:IFA$=""THEN60
70 IFA$="N"THEN100
80 FORA=1TO7:PRINT"■PUNKTE TEIL";A;":";:INPUTHS(A)
90 POKE704+B,HS(A)AND255:POKE705+B,HS(A)/256:B=B+2:NEXT:B=0
100 FORI=832TO832+33:READA:POKEI,A:NEXT
110 DATA120,169,51,133,1,169,0,133,95,133,90,133,88,169,208,133,96,169,240
115 DATA133,89,169,224,133,91,32,191,163,169,55,133,1,88,96
120 SYS832:POKE850,160:SYS832
125 POKE56576,PEEK(56576)AND252
130 POKE53272,8
135 POKE648,192:POKE2,255
140 PRINT"■BITTE WARTEN!"
150 FORA=1TO42:READC:NEXT:GOSUB50000
190 B=0:FORA=1TO7:HS(A)=PEEK(704+B)+PEEK(705+B)*256:B=B+2:HS=HS+HS(A):NEXT
200 GOTO1000
997 :
998 REM **  TEIL 1  **
999 :
1000 GOSUB12000
1010 C=PEEK(56320)
1020 IF(CAND4)=0THENGOSUB1350:Z=Z-.1
1030 IF(CAND8)=0ANDP<239THENGOSUB1300:Z=Z-.1
1040 IF(CAND1)=0THENGOSUB1400:Z=Z-.2:GOTO1060
1045 IF(CAND16)=0ANDP=31THENGOSUB1100
1046 IF(CAND16)=0ANDP=239THENGOSUB1150
1050 IF(CAND2)=0ANDP=239THENSC(1)=SC:GOTO5000
1060 Z=Z-.1:PRINT"■■ZEIT:"INT(Z)"■  ■FUEL:"FU"■  ■SCORE:"SC"■ ":IFZ<11THEN5000
1070 GOTO1010
1098 :
1099 REM ** TANK AUFLADEN **
1100 IFBE=1THEN1050
1105 POKESI+24,15:POKESI+6,240:POKESI+1,100:POKESI+4,17
1120 POKEH1,32:IFH1<50112THENH1=H1+40
1130 BE=1
1140 GOSUB12800:POKEV+42,6:POKESI+4,0
1145 RETURN
1148 :
1149 REM ** TANK ABLADEN **
1150 IFBE=0THEN1050
1155 POKESI+24,15:POKESI+6,240:POKESI+1,10:POKESI+4,33
1160 SF=INT(1000/Z*20):SC=SC+SF:FU=FU+FG
1170 POKEH2,81:POKEH2+DI,6:H2=H2-40
1180 POKEV+42,1:BE=0:POKESI+4,0
1190 RETURN
1297 :
1298 REM ** LAUF-/SPRINGROUTINEN **
1299 REM RECHTS
1300 FORA=2TO8STEP2:POKEK,R(A):POKEV+6,P+A:GOSUB1500:NEXT:P=P+8:RETURN
1349 REM LINKS
1350 FORA=2TO8STEP2:POKEK,L(A):POKEV+6,P-A:GOSUB1500:NEXT:P=P-8:RETURN
1399 REM SPRINGEN RECHTS
```

```
1400 IFPEEK(K)>21ANDP>38THEN1450
1405 IFP>30ANDP<231THEN1410
1406 RETURN
1410 FORA=2TO8STEP2:POKEK,R(A):POKEV+6,P+A*2:POKEV+7,Q-Q(A):GOSUB1500
1420 NEXT:P=P+16:RETURN
1449 REM SPRINGEN LINKS
1450 FORA=2TO8STEP2:POKEK,L(A):POKEV+6,P-A*2:POKEV+7,Q-Q(A):GOSUB1500
1460 NEXT:P=P-16:RETURN
1499 REM TON LAUFEN / KOLLISION?
1500 IFDU=3ORA=2ORA=6THENRETURN
1510 POKESI+1,07:POKESI+24,15:POKESI+4,33:POKESI+6,240:POKESI+4,0
1515 IFPEEK(V+31)=8THEN1550
1520 RETURN
1548 :
1549 REM ** AUF MINE GETRETEN **
1550 IFPEEK(V+7)<200THEN1520
1560 IFPEEK(V+6)<39THEN1700
1570 POKEK,0:POKEINT(((PEEK(V+6)-20)/8)+50112),32:POKEV+42,2
1580 POKESI+24,15:POKESI+1,10:POKESI+6,0:POKESI+4,0:POKESI+5,12:POKESI+4,129
1590 PRINT"▒▒▒▒▒▒▒▒▒■▒NA ENDLICH! NUN HABEN"
1600 PRINT"▒■■SIE EINE MINE GETROFFEN!"
1610 PRINT"▒■■HAR, HAR, ICH GRATULIERE!"
1620 PRINT"▒■■IHRE PUNKTEZAHL VER-"
1630 PRINT"▒■■FAELLT ZU GUNSTEN DER"
1640 PRINT"▒■■AKTION SORGENKIND!":POKE198,0
1650 FORW=1TO100:NEXT:POKEV+6,0:POKEV+7,0
1660 POKE198,0:WAIT198,1:RUN
1698 :
1699 REM ** IN TANKS GESPRUNGEN **
1700 POKESI+24,15:POKESI+1,10:POKESI+6,0:POKESI+4,0:POKESI+5,14:POKESI+4,129
1710 FORA=0TO40:POKE53281,7:FORW=1TO25:NEXT:POKE53281,0:FORW=1TO25:NEXT:NEXT
1720 POKE53281,14:PRINT"▒■WIE KANN MAN NUR SO BLOED SEIN UND IN"
1730 PRINT"▒DIE TANKS SPRINGEN? DADURCH HABEN SIE"
1740 PRINT"▒EINE EXPLOSION AUSGELOEST, DIE DIE GANZE"
1750 PRINT"ANLAGE ZERSTOERT HAT!"
1760 PRINT"▒▒WENN SIE NOCH EINMAL SPIELEN WOLLEN,"
1770 PRINT"▒MUESSEN SIE SCHON 'RUN' EINGEBEN."
1780 POKEV+21,0
1790 END
1997 :
1998 REM **  TEIL 2+4  **
1999 :
2000 GOTO10000:REM ERKLAERUNG
2010 POKESI,3:POKESI+1,17:POKESI+7,3:POKESI+8,16
2020 POKESI+5,32:POKESI+6,255:POKESI+12,32:POKESI+13,255: POKESI+15,200
2030 POKESI+20,80:POKESI+18,129:POKESI+24,15
2040 POKESI+4,17:POKESI+11,17:WI=200:POKEV+31,0
2045 IF(PEEK(56320)AND16)<>0THEN2045
2050 A=PEEK(56320):IFFU<1THENFU=0:GOTO2085
2060 IF(AAND4)=0THENP=P-2:FU=FU-4
2070 IF(AAND8)=0ANDP<253THENP=P+2:FU=FU-4
2080 IF(AAND16)=0THENSU=SU-.2:FU=FU-20:A=16
2081 IFA=16THENPOKESI+11,17:POKESI+8,7:POKESI+1,2:POKESI+4,129:GOTO2090
2085 POKESI+11,17:POKESI+8,16:POKESI+1,17:POKESI+4,17
2090 W=INT(RND(TI)*2):IFW=1ANDWI<245THENWI=WI+10:POKESI+15,WI:GOTO2100
2095 IFWI>99THENWI=WI-10:POKESI+15,WI
2100 SU=SU+.05:Q=Q+SU:IFQ<95THEN2110
2105 W=INT(RND(TI)*3):P=P-W
2110 POKEV+10,P:POKEV+11,Q
2115 PRINT"▒HOEHE:"205-INT(Q)"■ ":PRINT"▒"TAB(12)"▒SPEED:"INT(SU*10+.5)/10"■ "
2116 PRINT"▒"TAB(25)"▒FUEL:"FU"■ ▒▒"
2120 IFPEEK(V+31)=32ORQ<10THEN2500
2125 IFDU=2ANDPEEK(V+11)=59ANDP<150THEN2150
2130 IFQ<205THEN2050
2140 IFINT(SU*10+.5)/10>LGTHEN2500
2145 IFDU=2THEN2050
2148 :
2149 REM ** MONDLANDUNG ERFOLGREICH **
2150 POKESI+11,0:POKESI+18,0:FORA=1TO5
2160 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒KONTAKT!":FORW=1TO400:NEXT:POKESI+1,10:POKESI+4,17
2170 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒KONTAKT!":FORW=1TO400:NEXT:POKESI+4,0:NEXT
```

```
2180 PRINT"SQQQQQQQQQQ          "
2185 IFDU=2THEN2650
2190 PRINT"ːT]DAS HABEN SIE HUEBSCH GEMACHT. SIE ER-"
2195 IFSU=0THENSF=3000:GOTO2200
2197 SF=INT((1.2-INT(SU*100-.5)/100)*2500)
2200 PRINT"QHALTEN FUER DIESE GELUNGENE LANDUNG"
2205 PRINT"QQ"SF"PUNKTE. GRATULIERE!"
2210 SC(2)=SF:SC=SC+SC(2):PRINT"QDAMIT HABEN SIE NUN";SC;"PUNKTE."
2215 PRINT"QQHIGHSCORE:"HS(2)"PUNKTE"
2220 PRINT"QQWEITER: KNOPF DRUECKEN!"
2230 GOSUB15000
2240 GOSUB3000: REM TEIL3
2298 :
2299 REM **  TEIL 4  **
2300 IFUN=1THEN2335
2305 PRINT"ːQQNUN MUESSEN SIE WIEDER ZUR RAUMBASIS ZU-"
2310 PRINT"RUECKKEHREN. SEIEN SIE AEUSSERST VOR-"
2320 PRINT"QSICHTIG. JEDE BERUEHRUNG LOEST UNWEIGER-"
2330 PRINT"LICH EINE EXPLOSION AUS."
2335 PRINT"ːQQWEITER: KNOPF DRUECKEN"
2340 GOSUB15000
2350 POKEV+21,33:POKEV+11,205:POKEV+10,176
2355 P=176:Q=205:DU=2
2360 POKE53280,11:POKE53281,11:GOTO10180
2498 :
2499 REM ** EXPLOSION: RAUMBASIS **
2500 POKESI+11,0:POKESI+18,0
2505 POKESI+24,15:POKESI+1,10:POKESI+6,0:POKESI+4,0:POKESI+5,14:POKESI+4,129
2510 IFFU<=0ORQ>150THEN2550
2520 FORA=0TO40:POKE53281,7:FORW=1TO25:NEXT:POKE53281,0:FORW=1TO25:NEXT:NEXT
2530 PRINT"ː]SIE UNGLUECKLICHER! SIE SIND MIT DER":POKEV+21,0
2532 PRINT"QRAUMBASIS KOLLIDIERT UND HABEN EINE GAR"
2535 PRINT"QFUERCHTERLICHE EXPLOSION AUSGELOEST."
2536 PRINT"QSIE HABEN ZWAR UEBERLEBT, DOCH SCHWEBEN"
2537 PRINT"QHILFLOS IN DEN WEITEN DES ALLS.":POKE53280,11:POKE53281,11
2538 PRINT"QICH SCHAETZE IN WENIGEN MINUTEN GEHT"
2539 PRINT"QIHNEN DER SAUERSTOFF AUS. SIE TUN MIR"
2540 PRINT"QRICHTIG LEID. HAR, HAR!"
2541 IFGR<6THENPOKE198,0:WAIT198,1:RUN
2542 PRINT"QQABER SIE HABEN GLUECK. ES KOMMT EINE"
2543 PRINT"QGUTE FEE VORBEI UND LAESST SIE ES MIT":PRINT"Q2000 PUNKTEN WENIGER";
2544 PRINT"NOCH EINMAL PRO-":PRINT"QBIEREN! QKNOPF DRUECKEN!"
2545 GOSUB15000
2546 SC=SC-2000:IFDU=2THEN2350
2547 FU=FV:GOTO10170
2548 :
2549 REM ** EXPLOSION FAEHRE **
2550 POKE50173,0:FORW=1TO300:NEXT:POKEV+21,1:POKE50173,26
2555 IFFU<=0THEN2600
2560 PRINT"SQQQQQQQQQQQDIE FAEHRE IST JA WIE EINE BOMBE EINGE-"
2570 PRINT"QSCHLAGEN. NA JA, ICH WILL JA NICHT SO"
2580 PRINT"QSEIN. SIE DUERFEN ES NOCH EINMAL PRO-"
2590 PRINT"QBIEREN. ALLERDINGS MUSS ICH IHNEN 2000"
2593 PRINT"QPUNKTE ABZIEHEN. KNOPF DRUECKEN!":SC=SC-2000
2595 GOSUB15000
2596 IFDU=2THEN2350
2597 FU=FV:GOTO10170
2598 :
2599 REM ** TREIBSTOFFMANGEL **
2600 POKEV+21,1
2610 PRINT"SQQQQQQQQQQQIST IHNEN WOHL DER TREIBSTOFF AUSGE-"
2620 PRINT"QGANGEN? WIE SCHADE! TJA, SIE HABEN EBEN"
2630 PRINT"QAM ANFANG ZU WENIG MIT AUF DIE REISE GE-"
2640 PRINT"NOMMEN! AM BESTEN BEGINNEN SIE NOCH"
2645 PRINT"QEINMAL VON VORN!"
2646 POKE198,0:WAIT198,1:RUN
2648 :
2649 REM ** BASISLANDUNG ERFOLGREICH **
2650 PRINT"ːT]SEHR GUT. DIESE LANDUNG WAR VORZUEGLICH."
2655 IFSU=0THENSF=3000:GOTO2670
2660 SF=INT((1.2-INT(SU*100-.5)/100)*2500)
```

```
2665 PRINT"SIE BRINGT IHNEN";SF;"PUNKTE EIN."
2670 SC=SC+SF:PRINT"DAMIT HABEN SIE";SC;"PUNKTE.":SC(4)=SF
2675 PRINT"HIGHSCORE:"HS(4)"PUNKTE"
2680 PRINT"WEITER: KNOPF DRUECKEN"
2690 GOSUB15000
2700 PRINT"DA SIE DIE MISSION ERFUELLT HABEN        "
2710 PRINT"DUERFEN SIE NUN DIE RUECKREISE ANTRETEN.";
2720 FORA=1TO3:PRINT"                                      ":NEXT
2730 FORW=1TO3000:NEXT
2740 POKESI+4,129 :POKEV+21,49:POKEV+8,48:POKEV+9,80
2750 FORA=59TO0STEP-1:POKEV+9,A+21:POKEV+1,A:FORW=1TO200:NEXT:NEXT
2760 FORA=15TO0STEP-.01:POKESI+24,A:NEXT
2770 GOTO4000
2797 :
2798 REM **  TEIL 3  **
2799 :
3000 GOTO11000: REM ERKLAERUNG
3100 C=PEEK(56320):IFC=127THEN3100
3110 IF(CAND1)=0THENGOSUB3500
3120 IF(CAND2)=0THENGOSUB3550
3130 IF(CAND8)=0ANDP<248THENGOSUB1300
3140 IF(CAND4)=0ANDP>23THENGOSUB1350
3150 IF(CAND16)=0THEN3300
3155 IF(CAND16)=0THEN3450
3160 O2=O2-L:PRINT""TAB(34)"O2":PRINT""TAB(32)INT(O2)" "
3165 IFO2<1THEN3400
3170 X=INT((P-23)/8):Y=INT((Q-36)/8):X1=X-6:X2=X+6:Y1=Y-6:Y2=Y+6
3180 IFUX<X2ANDUX>X1ANDUY<Y2ANDUY>Y1THEN3200
3185 POKESI+7,36
3190 IFPEEK(49152+X+40*Y)>65THEN3250
3195 GOTO3100
3199 REM ENTFERNUNG DES URANS
3200 EX=ABS(X-UX):EY=ABS(Y-UY):E=INT(SQR(EX*EX+EY*EY)+.5):IFE>6THENE=6
3210 POKESI,36*(7-E):PRINTTAB(32)""A$(6-E):GOTO3190
3248 :
3249 REM ** IN KRATER GEFALLEN **
3250 POKESI+4,17:FORA=100TO0STEP-1:POKESI+1,A:NEXT:POKESI+4,0
3260 POKEV+21,0:PRINT"SIE SIND IN EINEN KRATER GEFALLEN UND"
3270 PRINT"HABEN SICH DEN FUSS VERSTAUCHT. DESHALB"
3275 PRINT"VERBRAUCHEN SIE VON NUN AN DOPPELT SO"
3280 PRINT"VIEL SAUERSTOFF!":L=L*2
3285 PRINT"WEITER: KNOPF DRUECKEN"
3287 GOSUB15000
3290 P=151:Q=140:POKEV+6,P:POKEV+7,Q:POKEV+21,40
3295 GOTO11430
3298 :
3299 REM ** URAN GEFUNDEN **
3300 IFX<>UXORY<>UYTHEN3155
3305 POKESI+4,17:FORA=0TO100:POKESI+1,A:NEXT
3310 FORA=100TO0STEP-1:POKESI+1,A:NEXT:POKESI+4,0:UR=UR+1:O2=O2+20
3315 SF=INT(5000/O2*20):SC(3)=SC(3)+SF
3320 PRINT""TAB(33)"URAN":PRINT""TAB(33)UR" "
3325 PRINT""TAB(32)"SCORE":PRINT""TAB(32)SC(3)" "
3330 PO=49152+40*UY+UX:POKEPO,65:POKEDI+PO,7:GOTO11500
3397 :
3398 :
3399 REM ** KEIN SAUERSTOFF **
3400 PRINT"TJA, IHNEN IST WOHL DER SAUERSTOFF AUS-"
3410 PRINT"GEGANGEN. ICH WUENSCHE IHNEN FROEHLICHES"
3420 PRINT"ERSTICKEN! DAMIT IST DIESES SPIEL FUER"
3430 PRINT"SIE BEENDET."
3435 POKESI+4,33:FORA=200TO20STEP-5:FORB=ATOA-20STEP-1:POKESI+1,B:NEXT
3440 FORB=A-20TOA:POKESI+1,B:NEXT:NEXT:POKESI+4,0
3445 POKE198,0:WAIT198,1:RUN
3448 :
3449 REM ** URANSUCHEN BEENDET **
3450 IFP<115ORP>155ORQ<>140THEN3100
3455 POKEV+21,0:POKESI+4,0:PRINT"OK! AUSGEZEICHNET!";UR;"URANVOR-"
3460 PRINT"KOMMEN SIND NICHT SCHLECHT. SIE BRINGEN"
3465 SC=SC+SC(3)
3470 PRINT"IHNEN"SC(3)"PUNKTE. DAMIT HABEN SIE"
```

```
3480 PRINT"■■"SC"PUNKTE. IMMERHIN!"
3490 PRINT"■■HIGHSCORE:"HS(3)"PUNKTE"
3495 FORA=1TO2000:NEXT:RETURN
3498 :
3499 REM ** LAUFEN: SCHRAEG OBEN/UNTEN **
3500 IFQ<45THENRETURN
3502 IFPEEK(K)>21THEN3530
3505 IFP>247THENRETURN
3510 FORA=2TO8STEP2:POKEK,R(A):POKEV+7,Q-A:POKEV+6,P+A:GOSUB1500
3520 NEXT:P=P+8:Q=Q-8:RETURN
3530 IFP<24THENRETURN
3535 FORA=2TO8STEP2:POKEK,L(A):POKEV+7,Q-A:POKEV+6,P-A:GOSUB1500
3540 NEXT:P=P-8:Q=Q-8:RETURN
3550 IFQ>227THENRETURN
3552 IFPEEK(K)>21THEN3580
3555 IFP>247THENRETURN
3560 FORA=2TO8STEP2:POKEK,R(A):POKEV+7,Q+A:POKEV+6,P+A:GOSUB1500
3570 NEXT:P=P+8:Q=Q+8:RETURN
3580 IFP<24THENRETURN
3585 FORA=2TO8STEP2:POKEK,L(A):POKEV+7,Q+A:POKEV+6,P-A:GOSUB1500
3590 NEXT:P=P-8:Q=Q+8:RETURN
3998 REM **  TEIL 5  **
3999 :
4000 GOSUB13000
4100 C=PEEK(56320)
4110 IF(CAND4)=0ANDP>23THENP=P-3
4120 IF(CAND8)=0ANDP<250THENP=P+3
4130 IF(CAND16)=0THENGOSUB4200
4140 IFFA=1ANDBE>.4THENBE=BE-.05:T=T-5:POKESI+1,T
4150 POKEV+12,P:POKEV+13,Q
4160 Q=Q+BE:IFQ>228THEN4300
4170 A=RND(TI)*1.5:P=P-A
4180 IFP<2THEN4400
4190 GOTO4100
4198 :
4199 REM ** FALLSCHIRM OEFFNEN **
4200 POKESI+11,0:POKESI+8,9:POKESI+13,248:POKESI+11,33
4210 POKE50174,27:POKESI+4,128:FA=1:POKESI+11,32:RETURN
4298 :
4299 REM ** LANDUNG GESCHAFFT **
4300 IFBE>.4THEN4350
4310 POKESI+4,0:POKESI+11,0:POKESI+11,33:POKE50174,28:POKESI+11,32
4320 SF=INT((141-ABS(P-141))/.05):SC(5)=SF:SC=SC+SF
4330 FORW=1TO3000:NEXT:GOTO6000
4348 :
4349 REM ** SCHIRM ZU SPAET GEOEFFNET **
4350 POKESI+1,10:POKESI+6,0:POKESI+4,0:POKESI+5,12:POKESI+4,129
4360 POKEV+21,0:PRINT"■■■■■■■■■■ SIE HABEN DIE REISSLEINE ZU SPAET BE-"
4370 PRINT"■ TAETIGT! BEIM AUFSCHLAG HABEN SIE SICH"
4380 PRINT"■ DESHALB LEIDER ALLE KNOCHEN GEBROCHEN."
4390 PRINT"■ IHRE SCHWER ERKAEMPFTEN PUNKTEN SIND"
4395 PRINT"■ HIERMIT ALLE VERLOREN.":POKE198,0:WAIT198,1:RUN
4398 :
4399 REM ** AUSSERHALB GELANDET **
4400 POKESI+6,240:POKESI+4,129:FORA=50TO0STEP-1:POKESI+1,A:NEXT
4405 PRINT"■■■■■■■■■■ SIE SIND AUSSERHALB DES ZIELGEBIETS IN"
4410 PRINT"■ EINEM MOOR GELANDET. ZWAR KONNTEN SIE"
4420 PRINT"■ GERETTET WERDEN, DOCH MUSS ICH IHNEN"
4430 PRINT"■ FUER DIESE MISSERABLE LEISTUNG 2000":SC(5)=0:SC=SC-2000
4440 PRINT"■ PUNKTE ABZIEHEN!     ■KNOPF DRUECKEN!"
4450 GOSUB15000
4460 GOTO6000
4997 :
4998 REM ** START (ZU TEIL 1) **
4999 :
5000 FORA=50151TOH2STEP-40:POKEA,32:NEXT
5010 Z=INT(Z):IFZ<11THEN5400
5015 PRINT"■■■■■■■■■■■■■■HIGHSCORE:";HS(1);"PUNKTE"
5020 POKEK,19:POKEV+16,31:POKEV+6,0
5030 POKESI+24,15:POKESI+6,240:POKESI+4,17
5040 FORB=0TO50STEP.3:POKESI+1,B:NEXT
```

```
5050 FORA=228TO123STEP-.05:POKEV+7,A:NEXT
5060 FORB=50TO0STEP-.3:POKESI+1,B:NEXT:P=0
5070 FORB=0TO4:GOSUB1300:FORW=1TO50:NEXT:NEXT:POKEV+21,23
5080 FORA=2TO0STEP-1
5090 PRINT"■■ZEIT:"A"■ "
5100 IFA>10THEN5120
5110 POKESI+6,240:POKESI+4,33:POKESI+1,10:IFA=0THENPOKESI+1,50
5120 FORW=1TO300:NEXT:POKESI+4,0:FORW=1TO600:NEXT:NEXT
5130 POKEV+8,40:POKEV+9,225
5140 POKESI+4,129:POKESI+1,2:A=225:B=0
5150 A=A-B:POKEV+9,A:FU=FU-34:PRINT"■"TAB(10)"■FUEL:"FU"■ ":IFFU<0THEN5200
5160 GOSUB5300:B=B+.005:IFA>5THEN5150
5170 FORA=15TO0STEP-.01:POKESI+24,A:NEXT
5180 IFVE=1THENPOKE198,0:WAIT198,1:RUN
5198 :
5199 REM ** ZU WENIG TREIBSTOFF **
5200 POKEV+21,7:POKESI+4,0
5210 A=A-B:IFA<5THENB=-1:A=5
5215 POKEV+9,A:GOSUB5300:B=B-.01:IFA<225THEN5210
5220 POKE50168,0:POKE50169,1:POKE50170,2
5230 POKESI+1,10:POKESI+6,0:POKESI+4,0:POKESI+5,14:POKESI+4,129:POKEV+21,0
5240 FORA=0TO40:POKE53281,7:FORW=1TO25:NEXT:POKE53281,0:FORW=1TO25:NEXT:NEXT
5250 POKE50168,16:POKEV+21,0
5260 PRINT"■■MIT SO WENIG TREIBSTOFF KANN KEINE"
5270 PRINT"■RAKETE STARTEN. DAS NAECHSTE MAL SOLLTEN"
5280 PRINT"SIE MEHR MITNEHMEN!"
5290 POKE198,0:WAIT198,1:RUN
5299 REM UP: BEWEGUNG DER RAKETE
5300 IFA>42THENPOKEV+5,A-42
5310 IFA>84THENPOKEV+3,A-84
5320 IFA>126THENPOKEV+1,A-126
5330 RETURN
5398 :
5399 REM ** START VERPASST **
5400 PRINT"■■■■■■■■■■■ICH FUERCHTE SIE SCHLAEFER"
5410 PRINT"■■■HABEN DEN START IHRER"
5420 PRINT"■■■EIGENEN RAKETE VERPENNT!":VE=1:GOTO5080
6000 POKE53280,13:POKE53281,3:POKEV+21,0:B=0
6010 PRINT"■■"TAB(10)"AUSWERTUNG":PRINTTAB(10)"──────────"
6015 PRINT"■IHR ERGEBNIS:"TAB(25)"HIGHSCORE:"
6020 PRINT"■■TANKEN:    "SC(1)"PUNKTE"
6030 PRINT"■LANDUNG1:  "SC(2)"PUNKTE"
6040 PRINT"■URAN:      "SC(3)"PUNKTE"
6050 PRINT"■LANDUNG2:  "SC(4)"PUNKTE"
6060 PRINT"■NOTSCHIRM:"SC(5)"PUNKTE"
6065 IFGR<>0THENSC(6)=INT((FU/(GR*6000))*150):GOTO6070
6067 SC(6)=FU
6070 PRINT"■■■REST FUEL:"SC(6)"PUNKTE":SC=SC+SC(6)
6080 SC(7)=(10-GR)*1000:SC=SC+SC(7):SC(0)=SC
6085 PRINT"■STUFE"10-GR":"SC(7)"PUNKTE"
6090 PRINT"■■■ENDPUNKTZAHL:"SC
6100 PRINT"■■■■":ZS=0
6110 FORA=1TO5:PRINT"■■"TAB(25)HS(A)"PUNKTE":NEXT
6115 PRINT"■■"TAB(25)HS(6)"PUNKTE":PRINTTAB(25)"■"HS(7)"PUNKTE"
6120 PRINTTAB(25)"■■"HS"PUNKTE■":GOSUB6500
6125 IFSC<HSTHEN6150
6130 PRINT"■NEUER HIGHSCORE!";:GOSUB6500:B=0
6140 FORA=1TO7:POKE704+B,SC(A)AND255:POKE705+B,SC(A)/256:B=B+2:NEXT
6150 PRINT"■     ■-> ■DRUCKEN (J/N)?"
6155 GETA$:IFA$=""THEN6155
6157 IFA$="J"THENGOSUB6200
6160 POKE198,0:WAIT198,1:RUN
6198 :
6199 REM ** DRUCKEN **
6200 OPEN4,4
6210 FORA=1TO7
6220 PRINT#4,"TEIL";A;":";"    ";SC(A);"PUNKTE"
6230 NEXT:CLOSE4,4:RETURN
6498 :
6499 REM ** MELODIE **
```

```
6500 POKESI+2,13:POKESI+3,15:POKESI+5,63:POKESI+6,9:RESTORE
6505 FORA=0TO33:READB:NEXT
6510 READX,Y,D
6520 IFY=-1THENRETURN
6530 POKESI+1,X:POKESI,Y:POKESI+4,65:FORT=1TOD:NEXT:POKESI+4,0:GOTO6510
6550 DATA17,103,250,21,237,250,26,20,400,21,237,100,26,20,250,29,69,250
6560 DATA26,20,250,0,0,250,21,237,250,26,20,250,29,69,1000,26,20,250,0,0,250
6570 DATA-1,-1,-1
9997 :
9998 REM ** VORBEREITUNG TEIL 2 **
9999 :
10000 POKE53280,11:POKE53281,11:POKEV+21,0:IFUN=1THEN10170
10010 PRINT"SIE BEWEGEN SICH NUN IN DEN WEITEN DES"
10020 PRINT"WELTALLS ......."
10030 PRINT"NA JA, WENN MAN ES GENAU NIMMT, EIGENT-"
10040 PRINT"LICH NUR IN DER UMLAUFBAHN UM DEN MOND."
10050 PRINT"DORT HABEN SIE AN EINE RAUMBASIS ANGE-"
10060 PRINT"DOCKT, VON DER AUS SIE EINE EXPEDITION"
10070 PRINT"ZUM MOND UNTERNEHMEN. DAZU GIBT ES EINE"
10080 PRINT"SPEZIELLE MONDLANDEFAEHRE, DIE SIE ABER"
10090 PRINT"MIT HOECHSTENS"LG"KM/H GESCHWINDIGKEIT"
10100 PRINT"LANDEN DUERFEN! BREMSSCHUB GEBEN SIE"
10110 PRINT"DURCH DEN FEUERKNOPF! NACH LINKS UND"
10112 PRINT"RECHTS MIT DEM JOYSTICK. KNOPF DRUECKEN"
10115 GOSUB15000
10120 PRINT"VORSICHT ES HERRSCHT EIN MAGNETSTURM,"
10130 PRINT"DER SIE NACH LINKS ABTREIBEN LAESST."
10140 PRINT"ACHTEN SIE DARAUF, DASS SIE INNERHALB"
10150 PRINT"DER MARKIERUNGEN LANDEN! DIE BASIS"
10152 PRINT"SELBST IST DURCH EINEN SCHUTZSCHIRM GE-"
10154 PRINT"SCHUETZT, MIT DEM SIE ABER NICHT KOLLI-"
10156 PRINT"DIEREN DUERFEN! DASSELBE GILT FUER DIE"
10158 PRINT"BASIS!!!      VIEL GLUECK"
10160 PRINT" START: KNOPF DRUECKEN"
10165 GOSUB15000
10170 POKEV+21,33:POKEV+11,59:POKEV+10,145:P=145:Q=59:FV=FU
10180 POKEV+39,14:POKEV+38,1:POKEV,49:POKEV+1,59:POKEV+16,0:POKEV+23,0:SU=0
10190 PRINT"‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ‾ ┐"
10200 PRINT"|                              |"
10210 PRINT"                               "
10220 PRINT" ______________              |"
10230 PRINT" \/\/\/\/\/\/\/\/\/\/\/\/\/"
10240 PRINT" \/\/\/\/\/\/\/\/\/\/\/\/\/     |"
10250 PRINT" ________________________    "
10255 PRINT"L _ _ _ _ _ _ _ _ _ _ _     _ _"
10260 FORX=1TO20:PRINT"";:NEXT
10280 PRINT"______________↑     ↑____________";
10290 PRINT"                              "
10300 FORA=SITOSI+24:POKEA,0:NEXT
10310 GOTO2010
10997 :
10998 REM ** VORBEREITUNG TEIL 3 **
10999 :
11000 IFUN=1THEN11235
11005 POKEV+21,0:PRINT"NUN BEGINNT DIE EIGENTLICHE AUFGABE"
11010 PRINT"DIESES PROJEKTS. SIE SOLLEN URANVORKOM-"
11020 PRINT"MEN FINDEN. DAZU DIENT IHNEN EIN GEIGER-"
11030 PRINT"ZAEHLER - DER COMPUTER. ER ERZEUGT EIN"
11040 PRINT"KNACKEN, DAS UMSO HOEHER IST, JE NAEHER"
11050 PRINT"SIE SICH BEI EINEM URANLAGER BEFINDEN."
11060 PRINT"OPTISCH WIRD DIESES KNACKEN DURCH EINEN"
11070 PRINT"ZEIGER UNTERSTUETZT. ERST WENN DIESER"
11080 PRINT"VOLL AUSSCHLAEGT, HABEN SIE ES GEFUNDEN"
11090 PRINT"UND KOENNEN DEN FEUERKNOPF DRUECKEN!"
11100 PRINT"BEACHTEN SIE ABER, DASS DIE REICHWEITE"
11110 PRINT"DES GEIGERZAEHLERS BEGRENZT IST!"
11120 PRINT"WEITER: KNOPF DRUECKEN"
11125 GOSUB15000
11130 PRINT"DIE STEUERUNG ERFOLGT MIT DEM JOYSTICK!"
11140 PRINT"BEACHTEN SIE ABER, DASS SIE NACH SCHRAEG"
11150 PRINT"OBEN BZW. UNTEN STEUERN KOENNEN!"
```

```
11160 PRINT"SIE KOENNEN NATUERLICH NICHT UNBEGRENZT"
11170 PRINT"SUCHEN. ACHTEN SIE AUF IHREN SAUERSTOFF-"
11180 PRINT"VORRAT. SIE MUESSEN AUF JEDEN FALL BEI"
11190 PRINT"DER RAUMFAEHRE SEIN UND DEN FEUERKNOPF"
11200 PRINT"DRUECKEN, WENN ER SICH DEM ENDE NEIGT!"
11205 PRINT"PRO URANFUND ERHALTEN SIE ALS BELOHNUNG"
11210 PRINT"20 EINHEITEN. MEIDEN SIE DIE KRATER!"
11220 PRINT"START: KNOPF DRUECKEN!"
11230 GOSUB15000
11235 A$(0)="      "
11240 A$(1)="          "
11250 A$(2)="          "
11260 A$(3)="          "
11270 A$(4)="          "
11280 A$(5)="          "
11290 A$(6)="          "
11410 P=151:Q=140:POKEV+21,40:POKEV+6,P:POKEV+7,Q:Q2=200:POKEV+42,1
11420 POKEV+10,130:POKEV+11,140:DU=3:L=1:POKEV+31,0:POKEV+27,0
11430 PRINT"":FORA=0TO22:PRINT"                                        "
11440 NEXT
11450 FORA=1TO10:GOSUB11600:PO=49152+X+40*Y:POKEPO,87:NEXT
11460 FORA=1TO5:GOSUB11600:PO=49152+X+40*Y
11470 POKEPO,85:POKEPO+1,73:POKEPO+40,74:POKEPO+41,75:NEXT
11480 POKE49683,30:POKE49689,30:POKE55827,14:POKE55833,14
11490 PRINT"":FORA=1TO4:PRINTTAB(12)"        ":NEXT
11495 PRINT""TAB(33)"HIGH":PRINTTAB(32)""HS(3)
11500 GOSUB11600:UX=X:UY=Y:IFPEEK(49152+UX+40*UY)<>32THEN11500
11510 IFUX>11ANDUX<18ANDUY>10ANDUY<15THEN11500
11530 PRINT""TAB(33)"URAN":PRINT""TAB(33)UR" "
11540 POKESI+24,15:POKESI+2,243:POKESI+3,31:POKESI,0:POKESI+1,0:POKESI+6,240
11550 POKESI+4,65:PRINT""TAB(32)"ZEIGER":PRINT""TAB(32)A$(1)
11560 PRINTTAB(32)"      "
11570 X=0:Y=0:GOTO3100
11600 X=INT(RND(TI)*28)+1:Y=INT(RND(TI)*22)+1:RETURN
11997 :
11998 REM ** VORBEREITUNG TEIL 1 **
11999 :
12000 PRINT"":POKE53280,14:POKE53281,14
12010 PRINT"
12020 PRINT"
12030 PRINT"
12040 PRINT"
12050 PRINT"
12060 PRINT"
12070 PRINT"
12080 PRINT"              !BY ACHIM MUELLER!"
12090 PRINT"
12100 POKESI+24,15:POKESI+6,240:POKESI+4,33
12110 FORA=0TO70STEP5:POKESI+1,A:PRINT:NEXT
12120 FORA=70TO0STEP-.5:POKESI+1,A:NEXT
12125 PRINT"BRAUCHEN SIE DIE ERKLAERUNGEN (J/N)?":POKE198,0
12126 GETA$:IFA$=""THEN12126
12127 IFA$="N"THENUN=1:GOTO12370
12130 PRINT"ENDLICH IST DER GROSSE TAG GEKOMMEN!"
12140 PRINT"IHRE SELBSTKONSTRUIERTE RAKETE IST"
12150 PRINT"STARTBEREIT. SIE BEGINNEN DEN COUNTDOWN"
12160 PRINT"UND WOLLEN IHRE RAKETE BETANKEN."
12170 PRINT"DOCH DA! IHR BOESER NACHBAR HAT AUS"
12180 PRINT"RACHE FUER UNZAEHLIGE SCHLAFLOSE NAECHTE"
12190 PRINT"(DER BAU DER RAKETE VERLIEF NICHT GERADE"
12200 PRINT"LEISE) MINEN GELEGT.   KNOPF DRUECKEN"
12210 GOSUB15000
12220 PRINT"DA SIE NUN DEN COUNTDOWN NICHT MEHR"
12230 PRINT"STOPPEN KOENNEN, MUESSEN SIE UEBER DIE"
12240 PRINT"MINEN SPRINGEN, UND SO DEN TREIBSTOFF"
12250 PRINT"ZU IHRER RAKETE TRANSPORTIEREN. DIE"
12260 PRINT"STEUERUNG ERFOLGT UEBER DEN JOYSTICK:"
12270 PRINT"LINKS   - NACH LINKS GEHEN (1 FELD)"
12280 PRINT"RECHTS  - NACH RECHTS GEHEN (1 FELD)"
12290 PRINT"OBEN    - SPRINGEN (2 FELDER)"
12300 PRINT"UNTEN   - TANKEN BEENDEN UND STARTEN"
```

```
12310 PRINT"KNOPF   - TANK AUF- BZW. ABLADEN"
12320 PRINT"AUF- BZW. ABLADEN KANN MAN NATUERLICH"
12330 PRINT"NUR AUF DEN WEISSEN FELDERN. BIS 10 SEC"
12340 PRINT"VOR DEM START MUSS DAS BETANKEN BEENDET"
12350 PRINT"SEIN! ZUM STARTEN AUF DAS RECHTE WEISSE"
12360 PRINT"FELD GEHEN!    KNOPF DRUECKEN"
12365 GOSUB15000
12370 PRINT" 0  1  2  3  4  5  6  7  8  9  10":A=55297
12372 PRINT"LEICHT  SCHWIERIGKEITSGRAD  SCHWER":FORW=1TO2000:NEXT
12374 C=PEEK(56320):POKEA,1
12376 IF(CAND4)=0ANDA>55299THENPOKEA,6:A=A-3
12378 IF(CAND8)=0ANDA<55325THENPOKEA,6:A=A+3
12380 IF(CAND16)<>0THEN12374
12382 GR=((A-55297)/3)
12384 PRINT"SCHWIERIGKEITSGRAD:"GR
12386 GR=10-GR
12388 FORA=1TO2000:NEXT
12390 LG=0.1*GR+0.1:FG=GR*150+1000
12400 PRINT"";
12420 FORA=0TO15:PRINTTAB(27)"  ":NEXT
12430 PRINTTAB(27)"  ":PRINTTAB(27)"  "
12440 PRINTTAB(27)"  "
12450 PRINT""TAB(31)""
12460 PRINT""TAB(31)""
12510 POKE50113,100:POKE56257,1:POKE50139,100:POKE56283,1
12550 FORA=49432TO50112STEP40:POKEA,81:POKEA+DI,6:NEXT
12555 H1=49432:H2=50151
12560 POKEV+39,15:POKEV+40,15:POKEV+41,15
12565 POKEV+42,1:POKEV+43,7:POKEV+44,1: REM FARBEN
12570 POKEV+21,31:REM EINSCHALTEN
12580 POKEV+23,23: POKEV+28,7: POKEV+37,12: POKEV+38,0: REM RAKETE
12590 POKE50168,16:POKE50169,17:POKE50170,18
12595 POKE50171,19:POKE50172,25:POKE50173,26: REM BLOCKS ZUWEISEN
12600 POKEV+16,23:POKEV+27,24
12605 POKEV,40:POKEV+2,40:POKEV+4,40:POKEV+1,99:POKEV+3,141
12610 POKEV+5,183:POKEV+6,31:POKEV+7,228
12615 Q=228:P=31:K=50171
12620 POKEV+31,0:POKE53280,5
12640 R(2)=20:R(4)=21:R(6)=20:R(8)=19
12650 L(2)=23:L(4)=24:L(6)=23:L(8)=22
12660 Q(2)=2:Q(4)=4:Q(6)=2:Q(8)=0
12670 PRINT"                  "
12680 PRINT"                            "
12690 PRINT"                            "
12700 PRINT"      ":Z=120:RETURN
12800 FORB=50115TO50136:A=INT(RND(1)*3):IFA=1THEN12815
12810 IFC=0THENPOKEB,46:POKEB+DI,0:C=1:GOTO12820
12815 C=0:POKEB,32
12820 NEXT:RETURN
12997 :
12998 REM VORBEREITUNG TEIL 5
12999 :
13000 POKEV+21,0:PRINT"";:IFUN=1THENPOKE198,1:GOTO13140
13005 PRINT"SIE BEFINDEN SICH NUN IM LANDEANFLUG"
13010 PRINT"ZUR ERDE. BIS JETZT LAEUFT ALLES PLAN-"
13020 PRINT"MAESSIG. ZUM AUSLOESEN DES BREMSFALL-"
13030 PRINT"SCHIRMS KNOPF DRUECKEN!"
13040 IF(PEEK(56320)AND16)<>0THEN13040
13050 PRINT"DER BREMSFALLSCHIRM OEFFNET SICH NICHT!"
13060 PRINT"SIE MUESSEN AUSSTEIGEN UND MIT IHREM"
13070 PRINT"NOTFALLSCHIRM LANDEN!"
13080 PRINT"DURCH DEN JOYSTICK KOENNEN SIE WAEHREND"
13090 PRINT"DES FALLENS OHNE FALLSCHIRM (!) NACH"
13100 PRINT"LINKS BZW. RECHTS LENKEN. DIE REISSLEINE"
13105 PRINT"BETAETIGEN SIE DURCH DEN FEUERKNOPF."
13110 PRINT"DA LEICHTER SEITENWIND HERRSCHT, SOLLTEN"
13120 PRINT"SIE ABER VORHER NACH RECHTS STEUERN, UM"
13130 PRINT"DIE MARKIERUNG (WEISS) EINIGERMASSEN ZU"
13135 PRINT"ZU TREFFEN."
13140 PRINT"WENN SIE ZUM AUSSTEIGEN"
13150 PRINT"BEREIT SIND, TASTE DRUECKEN!":POKE198,0
```

```
13370 POKESI+24,15:POKESI+6,240:POKESI+4,33
13380 FORB=0TO99:FORA=50TO150:POKESI+1,A:NEXT
13390 GETA$:IFA$=""THENNEXT
13395 B=99
13400 PRINT"[illegible]":POKE53280,2:POKE53281,14:GOSUB12670:BE=2:FA=0
13410 POKEV+21,64:POKE50174,28:POKEV+45,0
13420 POKEV+27,64:P=141:Q=0
13430 PRINT"[illegible]________________________[illegible]______________________[illegible]"
13440 POKESI+4,0:POKESI+6,0:POKESI+1,255:POKESI+6,175:POKESI+4,129
13600 POKESI+24,15:T=255:RETURN
15000 FORGG=1TO3000:NEXT
15010 IF(PEEK(56320)AND16)<>0THEN15010
15020 RETURN
49999 REM ** SPRITES **
50000 FORA=50176TO51071:POKEA,0:NEXT
50010 FORA=0TO12:READA1,A2
50020 FORI=A1TOA2:READB:POKEI,B:NEXT:NEXT
50030 RETURN
50079 REM RAKETENSPITZE
50080 DATA50176,50238,0,48,0,0,236,0,3,171,0,3,171,0,3,171,0,3,171,0,14,170,192
50090 DATA58,170,176,59,187,176,59
50100 DATA187,176,59,187,176,59,187,176,59,187,176,59,187,176,58,170,176,234
50110 DATA170,172,234,170,172,234,170,172,234,170,172,234,170,172,234,170,172
50119 REM RAKETE MITTELTEIL
50120 DATA50240,50302,250,170,188,238,170,236,235,255,172,234,170,172,234,170
50130 DATA172,255,255,252,213,85,92,255,255,252,213,85,92,255,255,252,213,85,92
50140 DATA255,255,252,213,85,92,255,255,252,213,85,92,255,255,252,234,170,172
50150 DATA234,170,172,250,170,188,238,170,236,235,255,172
50159 REM RAKETE UNTERTEIL
50160 DATA50304,50366,234,170,172,234,170,172,234,170,172,234,170,172,235
50170 DATA171,172,235,171,172,235,171,172,235,171,172,235,171,172,235,171,172
50180 DATA235,171,172,235,171,172,235,171,172,235,171,172,239,239,236
50190 DATA48,48,48,48,48,48,220,220,220,220,220,220,220,220,220,220,220,220
50199 REM MANN R PHASE I
50200 DATA50392,50428,12,0,0,12,0,0,8
50210 DATA0,0,28,0,0,28,0,0,28,0,0,28,0,0,28,0,0,12,0,0,12,0,0,12,0,0,12,0,0,12
50229 REM MANN R PHASE II
50230 DATA50456,50492,12,0,0,12,0,0,8,0,0
50240 DATA28,0,0,60,0,0,62,0,0,28,0,0,28,0,0,12,0,0,12,0,0,10,0,0,10,0,0,18
50259 REM MANN R PHASE III
50260 DATA50520,50556,12,0,0,12,0,0,8,0
50270 DATA0,60,128,0,95,0,0,156,0,0,28,0,0,28,0,0,12,0,0,10,0,0,9,0,0,17,0,0,33
50279 REM MANN L PHASE I
50280 DATA50584,50620,24,0,0,24,0,0,8,0,0,28,0,0,28,0,0,28,0,0,28,0,0,28,0,0
50290 DATA12,0,0,12,0,0,12,0,0,12,0,0,12
50309 REM MANN L PHASE II
50310 DATA50648,50684,48,0,0,48,0,0,16,0,0,56,0,0,252,0,0,60,0,0,56,0,0,56,0,0
50320 DATA24,0,0,24,0,0,40,0,0,40,0,0,36
50329 REM MANN L PHASE III
50330 DATA50712,50748,24,0,0,24,0,0,8,0,0,158,0,0,125,0,0,28,128,0,28,0,0
50335 DATA28,0,0,12,0,0,20,0,0,36,0,0,34,0,0,33
50339 REM FEUER
50340 DATA50752,50775,120,120,120,248,248,252,252,252,252,252,252,252,124,124
50350 DATA120,120,120,120,48,48,48,32,16,16
50359 REM MONDFAEHRE
50360 DATA50828,50878,2,0,0,2,0,0,2,0,0,20,16,0,8,16,0,21,255,0,226,0,128
50370 DATA4,56,64,4,68,64,4,68,64,2,68,128,1,69,0,2,254,128,4,0,64,8,0,32
50380 DATA8,0,32,28,0,112
50389 REM FALLSCHIRMSPRINGER
50390 DATA50880,50938,7,192,0,31,240,0,63,248,0,127,252,0,255,254,0,255,254,0
50400 DATA128,2,0,64,4,0,64,4,0,32,8,0,32,8,0,16,16,0,16,16,0,8,32,0,8,32,0
50410 DATA4,64,0,5,64,0,3,128,0,1,0,0,2,128
50419 REM SPRINGER OHNE SCHIRM
50420 DATA50995,51005,1,0,0,3,128,0,1,0,0,2,128


READY.
```



Diamonds

In diesem Spiel muß man mit seiner Spielfigur in einer dreistöckigen Höhle Diamanten aufsammeln. Als Behinderung stehen Felsbrocken und Löcher im Weg,

durch welche man fällt, wenn man nicht darüberspringt. Fällt man durch ein Loch im 3. und 2. Stock, landet man ein Stockwerk tiefer und verliert eine Figur. Fällt man durch ein Loch im 1. Stock, ist das Spiel vorbei. Die Felsbrocken kann man mit seiner Hakke „abbauen" indem man von der rechten Seite 10 mal durch Druck auf den Knopf von Paddle 1 auf den Stein einschlägt.
Die Diamanten sammelt man durch Darüberlaufen ebenfalls nur von der rechten Seite. Ab und zu steigen Luftballons in die Luft. Hält man sich daran fest, indem man sich an die Stelle setzt, wo der Luftballon aufsteigt, wird man ins nächste Stockwerk befördert. Aber Vorsicht! Manchmal passiert es, daß der Ballaon während dem Flug platzt. Dann fällt man in den 1. Stock und verliert eine Spielfigur.
Gesteuert werden kann mit Paddle O, wobei man jedoch Paddle 1 für die Hacke zur Hilfe nehmen muß. Gesprungen wird mit Knopf O. Der Einsatz eines Joysticks macht das Spielen komfortabel.
Das Programm führt eine Hiscore-Liste, in die sich bis zu 10 Namen eintragen lassen. Diese werden nach Punktzahlen sortiert.

Zur Eingabe:
1. Programm „DIAMONDS" (Hauptprogramm) eingeben und speichern.
2. Shapetables eingeben und mit „BSAVE DIAMONDS.S HAPE, A$6000, L$ 3A7" speichern.
3. Programm „HISCORES.OBJ" abschreiben, mit „RUN" starten.

Zum Starten:
Das das Programm die HGR-Seite benutzt und relativ lang ist, muß der Basic-Pointer durch „POKE 103,0 : POKE 104,64" in die HGR2-Seite gelegt werden.

ACHTUNG!!! Infolgedessen bedeutet der Befehl „HGR2" jetzt soviel wie „NEW"!!

Nach den POKE-Angaben, „NEW" und „RUN DIAMONDS" eingeben ...

```
8  TEXT : HOME : PRINT  CHR$ (4)"BLOADDIAMONDS.SHAPE": POKE 232,0: POKE 23
    3,96: ROT= 0: PRINT  CHR$ (4)"OPENHISCORES": PRINT  CHR$ (4)"READHISC
    ORES": INPUT NA%: FOR I = 1 TO NA%: INPUT NA$(I),PP%(I): NEXT : PRINT
     CHR$ (4)"CLOSE"
9  SCALE= 1: GOSUB 3000
10 M = 3:X = 50:PP% = 0:L = 0
20  HGR : HOME : FOR B1 = 1 TO 3: FOR A1 = 1 TO 28: HCOLOR= 1: DRAW 11 AT
    A1 * 10 - 10,B1 * 50:M%(A1,B1) = 11: NEXT A1,B1
30  FOR B1 = 1 TO 3: FOR I = 1 TO  INT (3 *  RND (1)) + 1:A1 =  INT (20 *
     RND (1)) + 6: HCOLOR= 0: DRAW 11 AT A1 * 10 - 10,B1 * 50: DRAW 11 AT
    A1 * 10,B1 * 50:M%(A1,B1) = 0:M%(A1 + 1,B1) = 0: NEXT I,B1
40 L = L + 1:D = L:B = L * 3:T% = 400: IF L > 5 THEN T% = T% + 100
45 A = 1:B% = 3:G% = 0: FOR I = 1 TO B:B1 =  INT (3 *  RND (1)) + 1
50 A1 =  INT (22 *  RND (1)) + 5: IF M%(A1,B1) <  > 11 THEN 50
60 M%(A1,B1) = 20: SCALE= 3: XDRAW 9 AT A1 * 10 - 10,B1 * 50 - 21: XDRAW 9
     AT A1 * 10 - 10,B1 * 50 - 20: XDRAW 9 AT A1 * 10 - 10,B1 * 50 - 19: NEXT

70  FOR B1 = 1 TO 3: FOR I = 1 TO D
80 A1 =  INT (22 *  RND (1)) + 3: IF M%(A1,B1) <  > 11 THEN 80
90  SCALE= 1:M%(A1,B1) = 8: XDRAW 8 AT A1 * 10 - 10,B1 * 50 - 10: NEXT I,B
    1
100  HCOLOR= 1: HPLOT 1,0 TO 279,0 TO 279,15 TO 1,15 TO 1,0: HCOLOR= 3
110  HPLOT 105,3 TO 100,3 TO 100,12 TO 105,12: HPLOT 106,4 TO 106,11: HPLOT
     109,3 TO 111,3: HPLOT 109,12 TO 111,12: HPLOT 110,3 TO 110,12: HPLOT
     114,12 TO 114,4: HPLOT 115,3 TO 119,3: HPLOT 120,4 TO 120,12: HPLOT 1
     15,7 TO 119,7: HPLOT 123,12 TO 123,3: HPLOT 126,7 TO 129,3 TO 129,12
120  HPLOT 123,3 TO 126,7
130  HPLOT 132,4 TO 132,11: HPLOT 138,4 TO 138,11: HPLOT 133,3 TO 137,3: HPLOT
     133,12 TO 137,12: HPLOT 141,12 TO 141,3 TO 147,12 TO 147,3: HPLOT 155
     ,3 TO 150,3 TO 150,12 TO 155,12: HPLOT 156,4 TO 156,11: HPLOT 160,3 TO
     165,3: HPLOT 160,7 TO 165,7: HPLOT 160,12 TO 165,12: HPLOT 159,4 TO 1
     59,6
140  HPLOT 166,8 TO 166,11: HPLOT 159,11: HPLOT 166,4
150  FOR I = 5 TO M * 15 STEP 15: XDRAW 1 AT I,1: NEXT
160 M = M - 1: IF M =  - 1 THEN 3500
170  FOR I = 1 TO 500: NEXT :MB% = 0:J% = 0: POKE 0,40: POKE 1,255: CALL 7
     71:X% = 10:A = 2:X = X - 15: XDRAW 1 AT X,1
180 X% = A * 10 - 10:S% = 1:W% = 3:L% = 3:H% = 4:Y% = B% * 50 - 17: XDRAW
     S% AT X%,Y%: XDRAW W% AT X%,Y% + 14
190 WO% = W%:XO% = X%:YO% = Y%:SO% = S%
200  IF  PDL (0) > 156 AND J% = 0 THEN  GOSUB 300: GOTO 220
```

```
210  IF  PDL  (Ø) < 10Ø AND J% = Ø THEN  GOSUB 40Ø
220 P% = M%( INT (A),B%)
230  IF P% = 8 AND  PDL (Ø) > 156 THEN PP% = PP% + 2ØØ: XDRAW 8 AT X%,Y% +
     7:G% = G% + 1: POKE Ø,3Ø: POKE 1,2Ø: CALL 771:M%(A,B%) = 11: IF G% =
     3 * D THEN 4ØØØ
240  IF P% > 19 AND P% < 3Ø THEN  XDRAW S% AT X%,Y%: XDRAW W% AT X%,Y% + 1
     4: CALL 79Ø: FOR I = 1 TO 5ØØ: NEXT : GOTO 16Ø
250  IF  PEEK ( - 16287) > 127 OR J% THEN  GOSUB 5ØØ
260  IF  PEEK ( - 16286) > 127 THEN  GOSUB 1ØØØ
270  IF BL% OR  INT (36Ø *  RND (1)) < 1Ø THEN  GOSUB 15ØØ
280 T% = T% - 1: VTAB 22: PRINT "     Time : "T%"       Score : "PP%"   ": IF
     T% = Ø THEN 35ØØ
290  GOTO 19Ø
300  IF X% + 5 > 274 THEN  RETURN
310 A = A + .5:X% = X% + 5: IF S% = 2 THEN S% = 1:W% = 3:L% = 3:H% = 4
320  IF W% = L% THEN W% = H%: GOTO 34Ø
330 W% = L%
340  XDRAW SO% AT XO%,YO%: XDRAW WO% AT XO%,YO% + 14: XDRAW S% AT X%,Y%: XDRAW
     W% AT X%,Y% + 14: POKE Ø,3Ø: POKE 1,2: CALL 771: IF M%( INT (A),B%) =
     Ø AND J% = Ø THEN 45ØØ
350  RETURN
400  IF X% - 5 < 6 THEN  RETURN
410 A = A - .5:X% = X% - 5: IF S% = 1 THEN S% = 2:W% = 5:L% = 5:H% = 6
420  IF W% = L% THEN W% = H%: GOTO 44Ø
430 W% = L%
440  XDRAW SO% AT XO%,YO%: XDRAW WO% AT XO%,YO% + 14: XDRAW S% AT X%,Y%: XDRAW
     W% AT X%,Y% + 14: POKE Ø,3Ø: POKE 1,2: CALL 771: IF M%( INT (A),B%) =
     Ø AND J% = Ø THEN 45ØØ
450  RETURN
500  IF J% = Ø THEN Y%(1) = Y% - 2:Y%(2) = Y% - 5:Y%(3) = Y% - 8:Y%(4) = Y
     % - 5:Y%(5) = Y% - 2:Y%(6) = Y%
505 J% = J% + 1:XO% = X%:YO% = Y%: IF J% = 7 THEN J% = Ø: RETURN
510 SO% = S%:WO% = W%:Y% = Y%(J%): IF  PDL (Ø) > 156 THEN X% = X% + 5:A =
     A + .5: GOTO 53Ø
520  IF  PDL (Ø) < 1ØØ THEN X% = X% - 5:A = A - .5
530  XDRAW SO% AT XO%,YO%: XDRAW WO% AT XO%,YO% + 14: XDRAW S% AT X%,Y%: XDRAW
     W% AT X%,Y% + 14: POKE Ø,Y%(J%): POKE 1,1Ø: CALL 771: RETURN
1000  IF  PDL (Ø) < 1ØØ OR S% = 2 THEN  RETURN
1010  ROT= 48: XDRAW 7 AT X% + 8,Y% + 14: POKE Ø,8: POKE 1,1Ø: CALL 771
1020  IF M%( INT (A) + 1,B%) < 2Ø OR M%( INT (A) + 1,B%) > 3Ø THEN  XDRAW
     7 AT X% + 8,Y% + 14: ROT= Ø: RETURN
1030 M%( INT (A) + 1,B%) = M%( INT (A) + 1,B%) + 1: IF M%( INT (A) + 1,B%)
      = 3Ø THEN  GOSUB 1Ø5Ø
1040  ROT= 48: XDRAW 7 AT X% + 8,Y% + 14: ROT= Ø: SCALE= 1: RETURN
1050 M%( INT (A) + 1,B%) = 11: ROT= Ø: SCALE= 3: XDRAW 9 AT ( INT (A) + 1)
      * 1Ø - 1Ø,B% * 5Ø - 21: XDRAW 9 AT ( INT (A) + 1) * 1Ø - 1Ø,B% * 5Ø -
     2Ø: XDRAW 9 AT ( INT (A) + 1) * 1Ø - 1Ø,B% * 5Ø - 19: SCALE= 1: RETURN

1500  IF BL% = Ø THEN BL% = 1:BX% =  INT (X% - 6Ø *  RND (1)) + 3Ø:BY% = 1
     6Ø: IF BX% < Ø OR BX% > 265 THEN BL% = Ø: GOTO 15ØØ
1510 AX% = BX%:AY% = BY%:BY% = BY% - 5: IF BY% < 2Ø THEN  XDRAW 1Ø AT AX%,
     AY%:BL% = Ø: RETURN
1520  XDRAW 1Ø AT AX%,AY%: XDRAW 1Ø AT BX%,BY%
1530  IF MB% THEN 157Ø
1540  IF BY% = 125 AND B% = 3 AND BX% - 5 < X% AND BX% + 5 > X% THEN MB% =
     1: GOTO 157Ø
1550  IF BY% = 7Ø AND B% = 2 AND BX% - 5 < X% AND BX% + 5 > X% THEN MB% =
     1: GOTO 157Ø
1560  RETURN
1570 OY% = Y%:Y% = BY% + 14: XDRAW S% AT X%,OY%: XDRAW W% AT X%,OY% + 14: XDRAW
```



```
S% AT X%,Y%: XDRAW W% AT X%,Y% + 14: IF BY% = 75 THEN B% = 2: XDRAW S
     % AT X%,Y%: XDRAW W% AT X%,Y% + 14: POKE 0,50: POKE 1,20: CALL 771:MB
     % = 0:J% = 0:BL% = 0: XDRAW 10 AT BX%,BY%: GOTO 180
1580  IF BY% = 20 THEN B% = 1: XDRAW S% AT X%,Y%: XDRAW W% AT X%,Y% + 14: POKE
     0,50: POKE 1,20: CALL 771:MB% = 0:J% = 0:BL% = 0: XDRAW 10 AT BX%,BY%
     : GOTO 180
1590  IF  INT (500 *  RND (1)) + 1 < 6 THEN 1610
1600  GOTO 1500
1610  CALL 790: XDRAW 10 AT BX%,BY%: FOR I = 1 TO 200: NEXT I: FOR I = 1 TO
     10: POKE 0,10: POKE 1,10: CALL 771: POKE 0,20: POKE 1,10: CALL 771: NEXT
     :Q = 0
1620  FOR I = Y% TO 160 STEP 5:YO% = Y%:Y% = I: XDRAW S% AT X%,YO%: XDRAW
     W% AT X%,YO% + 14: XDRAW S% AT X%,Y%: XDRAW W% AT X%,Y% + 14: POKE 0,
     I: POKE 1,5: CALL 771: NEXT : XDRAW S% AT X%,Y%: XDRAW W% AT X%,Y% +
     14:B% = 3:BL% = 0: CALL 790: FOR I = 1 TO 500: NEXT : GOTO 160
2990  END
3000  DIM M%(28,3)
3010  DATA 162,0,189,173,48,192,136,208,4,198,1,240,8,202,208,246,166,0,76
     ,3,3,96,162,1,138,32,168,252,141,48,192,232,224,50,240,3,76,24,3,96
3020  FOR I = 768 TO 807: READ J: POKE I,J: NEXT I
3025  FOR I = 1 TO 8: READ T(I),L(I): NEXT I
3026  DATA 127,99,127,99,103,128,127,128,111,64,103,64,84,100,63,255
3027  DIM T1(8),L1(8): FOR I = 1 TO 8: READ T1(I),L1(I): NEXT
3028  DATA 127,35,127,35,127,35,84,70,103,70,63,70,1,70,159,130
3030  VTAB 5: HTAB 4: PRINT "pear software games intersection": VTAB 7: HTAB
     12: PRINT "proudly presents": VTAB 11: HTAB 14: PRINT "------------":
      VTAB 13: HTAB 14: PRINT "------------": INVERSE : VTAB 12: HTAB 14: PRINT
     ": DIAMONDS :": NORMAL : VTAB 16: HTAB 13: PRINT "By  Mark Heene"
3040  VTAB 24: HTAB 1: PRINT "V3.0";: VTAB 23: HTAB 9: PRINT "Press button
      0 to play": FOR I = 1 TO 600: IF  PEEK ( - 16287) > 127 THEN  HOME :
      RETURN
3050  NEXT I: GOSUB 3060: HOME : GOTO 3030
3060  HOME : VTAB 5: INVERSE : PRINT "---- DIAMONDS TOP TEN HISCORE TABLE
     ----": NORMAL
3070  INVERSE : VTAB 8: HTAB 18 -  LEN (NA$(1)): PRINT NA$(1);: HTAB 20: PRINT
     PP%(1):HI% = PP%(1): NORMAL
3080  FOR I = 2 TO NA%: VTAB 8 + I: HTAB 18 -  LEN (NA$(I)): PRINT NA$(I);
     : HTAB 20: PRINT PP%(I): NEXT I
3090  VTAB 23: HTAB 7: PRINT "Press button 0 to continue": FOR I = 1 TO 10
     00: IF  PEEK ( - 16287) > 127 THEN  RETURN
3100  NEXT I: RETURN
3110  END
3500  FOR I = 1 TO 500: NEXT I: FOR I = 1 TO 8: IF T1(I) = 1 THEN  FOR J =
     1 TO L1(I) * 2: NEXT J: GOTO 3502
3501  POKE 0,255 - T1(I): POKE 1,L1(I): CALL 771
3502  NEXT
3505 MB% = 0:J% = 0:BL% = 0
3510  TEXT : HOME : VTAB 12: HTAB 15: PRINT "Game over": VTAB 15: HTAB 7: PRINT
     "Score : "PP%: VTAB 17: HTAB 7: PRINT "Level : "L
3520  VTAB 23: HTAB 7: PRINT "Press button 0 to continue";: WAIT  - 16287,
     128
3530  IF PP% >  = 1000 THEN 3550
3540  GOSUB 3060: WAIT  - 16287,128: GOTO 10
3550  HOME :NA% = NA% + 1: IF NA% = 11 AND PP% > HI% THEN NA% = 10: GOTO 3
     580
3560  IF NA% = 11 AND PP% > PP%(10) THEN NA% = 10: GOTO 3580
3570  IF NA% = 11 THEN NA% = 10: GOSUB 3060: WAIT  - 16287,128: GOTO 10
3575  IF PP% > PP%(1) THEN  VTAB 3: FLASH : PRINT "-- NEW HISCORE !!! --"
3580  INVERSE : VTAB 5: PRINT "ENTER YOUR NAME ->";: NORMAL : INPUT "";NA$
     (NA%):PP%(NA%) = PP%
```



```
3590 AA% = 1:ZZ% = NA%:BB% = 0:CC% = 1
3600  FOR I = AA% TO ZZ%: FOR J = AA% TO ZZ%
3610  IF PP%(I) < PP%(J) THEN AA% = J: GOTO 3600
3620  NEXT J:BB% = BB% + CC%:XX = PP%(I):X$ = NA$(I):PP%(I) = PP%(BB%):NA$
     (I) = NA$(BB%):PP%(BB%) = XX:NA$(BB%) = X$:AA% = BB% + CC%: IF AA% <
      = ZZ% THEN 3600
3630  NEXT I
3640  PRINT : PRINT  CHR$ (4)"OPENHISCORES": PRINT  CHR$ (4)"DELETEHISCORE
     S": PRINT  CHR$ (4)"OPENHISCORES": PRINT  CHR$ (4)"WRITEHISCORES": PRINT
     NA%: FOR I = 1 TO NA%: PRINT NA$(I): PRINT PP%(I): NEXT I: PRINT  CHR$
     (4)"CLOSE"
3650  GOSUB 3060: WAIT  - 16287,128: GOTO 10
4000  FOR I = 1 TO 300: NEXT I: FOR I = 1 TO 8: POKE 0,255 - T(I): POKE 1,
     L(I): CALL 771: NEXT I
4005 J% = 0:MB% = 0:BL% = 0
4007 PP% = PP% + T%
4010  TEXT : HOME : VTAB 12: HTAB 11: PRINT "Level "L" finished"
4015 M = M + 1:X = X + 15
4020  IF L / 2 =  INT (L / 2) THEN  VTAB 14: HTAB 5: FLASH : PRINT "EXTRA
     SAM !!!":X = X + 15:M = M + 1: IF M > 5 THEN M = M - 1:X = X - 15
4025  IF L + 1 > 10 THEN L = 0
4030  NORMAL : VTAB 16: HTAB 7: PRINT "Get ready to enter level "L + 1
4040  VTAB 23: HTAB 7: PRINT "Press button 0 to continue": WAIT  - 16287,1
     28: HOME : HGR : GOTO 20
4500  IF B% = 3 THEN 3500
4510  XDRAW S% AT X%,Y%: XDRAW W% AT X%,Y% + 14:B% = B% + 1: FOR I = 100 TO
     200 STEP 10: XDRAW 1 AT X%,Y% + Q: XDRAW 3 AT X%,Y% + Q + 14: POKE 0,
     I: POKE 1,10: CALL 771: XDRAW 1 AT X%,Y% + Q: XDRAW 3 AT X%,Y% + Q +
     14:Q = Q + 5: NEXT :Q = 0: CALL 790: GOTO 160
```

Shapes:

```
*6000.63A7

6000- 0B 00 18 00 A5 00 32 01
6008- 53 01 74 01 8D 01 AE 01
6010- DF 01 30 02 81 02 42 03
6018- 01 01 01 01 05 01 01 01
6020- 01 02 03 03 03 03 07 07
6028- 07 03 03 02 01 01 05 05
6030- 05 05 05 01 01 02 03 03
6038- 03 03 03 03 03 03 03 02
6040- 05 05 05 05 05 05 05 05
6048- 05 02 03 03 07 07 03 07
6050- 03 03 03 02 01 01 05 01
6058- 05 05 05 01 01 02 03 03
6060- 03 07 07 07 07 07 03 02
6068- 01 01 01 05 05 05 01 01
6070- 01 02 03 03 03 07 03 03
6078- 03 07 03 02 01 05 05 05
6080- 01 05 05 05 01 02 03 03
6088- 07 03 03 03 07 07 07 02
6090- 01 05 05 05 05 05 05 05
6098- 01 02 03 03 03 07 07 07
60A0- 07 07 03 02 00 01 01 01
60A8- 01 05 01 01 01 01 02 03
60B0- 03 03 03 07 07 07 03 03
60B8- 02 01 01 05 05 05 05 05
60C0- 01 01 02 03 03 03 03 03
60C8- 03 03 03 03 02 05 05 05
60D0- 05 05 05 05 05 05 02 03
60D8- 03 03 05 03 07 03 07 07
60E0- 02 01 01 05 05 05 01 05
60E8- 01 01 02 03 03 03 07 07
60F0- 07 07 07 03 02 01 01 01
60F8- 05 05 05 01 01 01 02 03
6100- 03 03 07 03 03 03 07 03
6108- 02 01 05 05 05 01 05 05
6110- 05 01 02 03 03 07 07 07
6118- 03 03 03 07 02 01 05 05
6120- 05 05 05 05 05 01 02 03
6128- 03 03 07 07 07 07 07 03
6130- 02 00 01 01 01 05 01 05
6138- 01 02 03 07 03 07 07 03
6140- 03 02 01 01 01 05 05 05
6148- 01 06 03 03 03 03 03 03
6150- 03 02 00 01 01 01 05 01
6158- 05 01 02 03 07 07 03 07
6160- 07 03 02 01 01 01 05 01
6168- 05 05 06 03 03 03 03 03
6170- 03 03 02 00 01 01 01 05
6178- 01 06 07 07 03 07 03 02
6180- 01 05 01 05 05 06 03 03
6188- 03 03 03 02 00 01 01 01
6190- 05 01 05 01 02 03 07 07
6198- 03 07 07 03 02 01 05 05
61A0- 05 01 05 01 02 03 03 03
61A8- 03 03 03 03 02 00 01 01
61B0- 01 05 01 01 01 02 03 03
61B8- 03 03 07 03 03 02 01 01
61C0- 01 05 01 01 01 02 03 03
61C8- 03 03 07 03 03 02 05 01
61D0- 01 05 01 01 05 02 03 03
61D8- 07 07 07 07 07 02 00 01
61E0- 01 01 01 05 01 01 01 01
61E8- 02 03 03 03 07 07 03 07
61F0- 07 03 02 05 05 01 01 05
61F8- 01 01 05 05 02 03 03 07
6200- 03 03 07 03 03 07 02 01
6208- 01 05 01 05 01 05 01 01
6210- 02 03 03 03 03 07 03 07
6218- 03 03 02 01 01 01 01 05
6220- 01 01 01 01 02 03 03 03
6228- 03 03 03 03 03 03 02 00
6230- 01 01 01 05 05 01 01 01
6238- 01 02 03 03 03 07 07 07
6240- 07 07 03 02 01 01 05 05
6248- 05 05 05 05 01 02 03 07
6250- 07 07 07 07 07 07 07 02
6258- 05 05 05 05 05 05 05 05
6260- 05 06 03 07 07 07 07 07
6268- 07 07 07 06 01 05 05 05
6270- 05 05 05 05 01 02 03 03
6278- 03 03 03 03 03 03 03 02
6280- 00 01 01 01 01 05 05 05
6288- 01 01 01 01 02 03 03 03
6290- 07 07 07 07 07 07 07 03
6298- 02 01 05 05 05 05 05 01
62A0- 01 05 05 01 02 03 07 07
62A8- 03 03 07 07 07 07 07 07
62B0- 06 05 05 05 05 05 05 05
62B8- 05 05 05 05 02 03 07 07
62C0- 07 07 07 07 07 07 07 07
62C8- 06 05 05 05 05 05 05 05
62D0- 05 05 05 05 02 03 07 07
62D8- 07 07 07 07 07 07 07 07
62E0- 06 01 05 05 05 05 05 05
62E8- 05 05 05 01 02 03 03 03
62F0- 07 07 07 07 07 07 07 03
62F8- 02 01 01 01 01 05 05 05
6300- 01 01 01 01 02 03 03 03
6308- 03 03 03 07 03 03 03 03
6310- 02 01 01 01 01 01 05 01
6318- 01 01 01 01 02 03 03 03
```




```
6320- 03 03 03 07 07 03 03 03
6328- 02 01 01 01 01 05 05 01
6330- 01 01 01 01 02 03 03 03
6338- 03 03 03 03 03 03 03 03
6340- 02 00 05 05 05 05 05 05
6348- 05 05 05 06 07 07 07 07
6350- 07 07 07 07 07 06 05 05
6358- 05 05 05 05 05 05 05 06
6360- 03 07 07 07 07 07 07 07
6368- 07 06 01 05 05 05 05 05
6370- 05 01 01 02 03 03 03 07
6378- 07 07 03 07 03 02 01 01
6380- 01 01 05 05 05 01 01 02
6388- 03 03 03 03 07 03 03 03
6390- 03 02 01 01 01 01 01 05
6398- 01 01 01 02 03 03 03 03
63A0- 07 03 03 03 03 02 00 00
```

Highscore Ersteller:

```
10  PRINT  CHR$ (4)"OPEN HISCORES"
20  PRINT  CHR$ (4)"WRITE HISCORES"
30  PRINT 2
40  PRINT "MARK": PRINT 1000: PRINT "APPLE": PRINT 800
50  PRINT  CHR$ (4)"CLOSE"
60  NEW
70  REM  DIESES PROGRAMM DIENT NUR ZUR ERSTELLUNG DES HIGHSCORE-FILES
```

Instswap

Das in Maschinensprache geschriebene Programm erweitert das Applesoft-Basic mit Hilfe des &-Zeichens um zwei Befehle: SWAP und INSTRING.

Es beginnt bei der Adresse Ü94DO und schützt sich durch Verändern des Himem-Wertes vor Überschreiben durch Basic-Variablen.
Gestartet wird das Programm mit „BRUN".

SWAP

Der SWAP-Befehl ermöglicht das Vertauschen von zwei Textvariablen ohne Zuhilfenahme einer dritten. Durch diese Methode wird die sog. „Garbage Collection" vermieden, wodurch z. B. Sortierprogramme sehr viel schneller werden können.
SYNTAX: & S, AÜ, BÜ
wobei für AÜ und BÜ beliebige Variablennamen stehen können. Stehen an Stelle der Textvariablen numerische, so erfolgt die Meldung „TYPE MISMATCH ERROR".
Beispiel:
ILIST
10 AÜ = „ABC"
20 BÜ = „XYZ"
30 & S, AÜ, BÜ
40 PRINT AÜ
50 PRINT BÜ
IRUN
XYZ
ABC
I

INSTRING

Diese Funktion sucht eine Zeichenkette in einer anderen und gibt bei erfolgreicher Suche die Position zurück.
SYNTAX: & I, AÜ, BÜ, P %
Der Text von BÜ wird in AÜ gesucht und die Position wird in P % wiedergegeben, wobei an Stelle von AÜ, BÜ und P % beliebige Variablen des entsprechenden Typs (Text- bzw. Ganzzahlvariablen) benutzt werden können. Ist ein Variablentyp falsch, so erfolgt ein „TYPE MISMATCH ERROR". Dieser erfolgt ebenfalls, wenn die erste Textvariable leer ist, d. h. die Länge null hat.
Beispiel:
ILIST
10 AÜ = „APPLE II"
20 BÜ = „PL"
30 & I, AÜ, BÜ, P %
40 PRINT P %
IRUN
3
I
In AÜ können auch „Schneckennudeln" als Jokerzeichen enthalten sein. Sie ersetzen kein, ein oder mehrere Zeichen.
Beispiel:
ILIST
10 AÜ = „APPLE II"
20 BÜ = „PÖE II"
30 & I, AÜ, BÜ, P %
40 PRINT P %
IRUN
2
I
Ist BÜ nicht in AÜ enthalten, so wird P % null.
Ü ≙ $
I ≙]
Ö ≙ e

Listing:

```
*94D0.95FD

94D0- A9 D0 85 73 A9 94 85 74
94D8- A9 4C 8D F5 03 A9 E8 8D
94E0- F6 03 A9 94 8D F7 03 60
94E8- 20 B7 00 E6 B8 D0 02 E6
94F0- B9 C9 53 F0 07 C9 49 F0
94F8- 21 4C C9 DE 20 E2 95 A5
9500- 83 85 FE A5 84 85 FF 20
9508- E2 95 A0 02 B1 83 AA B1
9510- FE 91 83 8A 91 FE 88 10
9518- F3 60 20 E2 95 A0 00 B1
9520- 83 F0 59 85 FC C8 B1 83
9528- 8D 73 95 8D 9F 95 C8 B1
9530- 83 8D 74 95 8D A0 95 20
9538- E2 95 A0 00 B1 83 F0 3C
9540- 85 FD C8 B1 83 8D 63 95
9548- 8D 89 95 8D CB 95 C8 B1
9550- 83 8D 64 95 8D 8A 95 8D
9558- CC 95 A2 00 A0 00 84 FE
9560- 84 FF B9 FF FF C9 40 D0
9568- 07 C8 C4 FD F0 0E D0 F2
9570- 84 FB DD FF FF F0 0A E8
9578- E4 FC 90 F6 A9 00 4C D4
9580- 95 86 EB C8 C4 FD F0 48
9588- B9 FF FF C9 40 D0 0A A9
9590- 01 85 FE 85 FF 84 F9 D0
9598- EA E8 E4 FC F0 DE DD FF
95A0- FF D0 0C A5 FE F0 DC 86
95A8- FA A9 00 85 FE F0 D4 48
95B0- A5 FF F0 11 A5 FE D0 09
95B8- A6 FA A4 F9 C8 A9 01 85
95C0- FE 68 4C 99 95 A6 EB A4
95C8- FB 68 B9 FF FF 4C 77 95
95D0- A6 EB E8 8A 48 20 EB 95
95D8- A0 00 98 91 83 C8 68 91
95E0- 83 60 20 F5 95 F0 01 60
95E8- 4C 76 DD 20 F5 95 D0 F8
95F0- A5 12 F0 F4 60 20 BE DE
95F8- 20 E3 DF A5 11 60
```



GARTEN



Ein fleißiger Gärtner hat seinen kleinen Garten für die Saat vorbereitet. Nun muß er die Saat ausbringen.

Er kann aber immer nur für eine Beetreihe Samen nehmen und aussäen. Nach der Saat muß er natürlich gießen. Bald schon erscheinen die ersten jungen Pflänzchen, die dann schnell ausgereift sind und mit dem Tragekorb eingesammelt werden können. Da sie aber sehr groß sind, kann der kleine Gärtner nur höchstens 5 Pflanzen auf einmal tragen. Manche Keimlinge wachsen recht langsam, aber hier kann der Gärtner mit ein wenig Dünger ins Gießwasser nachhelfen. Die ganze Gärtnerei wäre leicht zu schaffen, wenn da nicht ein Schwarm gefräßiger Vögel wäre. Diese haben einen Riesenappetit auf das Saatgut und die ausgewachsenen Pflanzen, mögen aber nicht die kleinen Keimlinge. Der Gärtner kann sie mit der Vogelrassel verscheuchen, wenn sie gerade im Boden nach der Saat suchen.

Gespielt wird mit dem Joystick.
Man geht mit dem Gärtner zu dem Gegenstand, den man aufnehmen will, und drückt den Aktionsknopf. Ausgesät, gegossen, geerntet und die Ernte in den Vorratsbehälter geworfen geschieht ebenfalls durch Betätigen des Aktionsknopfes.
Nun viel Spaß und eine reichliche Ernte.

```
0 REM GARTEN *H.LOEPP*
10 PRINTCHR$(147)SPC(225)"EINEN MOMENT":GOSUB450
20 A$="▮▮CCCCCCCCC▯"
30 BS=7680:BZ=8186:FS=30720:FZ=38906:D=7704:C=7726
40 Z=1:SE=80:T=36877
50 PRINTCHR$(147):POKET+1,15:POKE36879,29:POKE36869,255
60 FORN=BSTOBS+21:POKEN,3:POKEN+440,3
65 POKEFS+N,5:POKEFS+N+440,5:NEXT
70 FORN=BS+22TOBS+440STEP22:POKEN,3:POKEN-1,3
75 POKEFS+N-1,5:POKEFS+N,5:NEXT
80 POKEBS+12,16:POKEBS+13,15:POKEBS+14,17:POKEBS+15,16
85 PRINTCHR$(19),CHR$(17):FORN=1TO9:PRINTA$:NEXT
90 PRINTCHR$(19)SPC(20)"▯▯B▯▯▯▮U▯▯▯▮E▯▯▯▮F▯▯▯▮K▯▯▯▯▮▮ST▯▮TT
▯▮P"
100 SYS828:J=PEEK(251):C1=PEEK(D+22):C2=PEEK(D+1)
105 POKET,0
110 IFJ+PEEK(D-22)=184THEND=D-22:POKED+22,32:POKET,254
115 IFJ+PEEK(D-1)=172THEND=D-1:POKED+1,32:POKET,254
120 IFJ+PEEK(D+22)=180THEND=D+22:POKED-22,32:POKET,254
125 IFJ+PEEK(D+1)=60THEND=D+1:POKED-1,32:POKET,254
130 IFRND(1)<.4THENGOSUB160
140 IFPEEK(D+22)=4ANDZ=8THENPOKED+22,13
145 IFPEEK(252)=0THENGOSUB250
150 POKED,Z:POKED+FS,0:GOTO100
160 IFPEEK(C)=12THENPOKEC,0:AN=AN-1:GOSUB410
165 IFPEEK(C)=4THENPOKEC,0:AN=AN-1:GOSUB410
170 IFPEEK(C)=13THENPOKEC,4:GOSUB400
175 IFPEEK(C)=14THENPOKEC,12
200 C=7725+INT(RND(1)*9)*44+INT(RND(1)*8)+1
210 IFAN=0ANDBL=0ANDSE=0ANDSA=0THEN350
220 RETURN
250 C1=PEEK(D+22):C2=PEEK(D+1)
255 IFC1=12ANDZ=7ANDBL<5THENPOKED+22,18:AN=AN-1:BL=BL+1:SN=S
N+1:PRINT"▯▮▮V"BL
260 IFC1=14ANDDU=1ANDZ=9THENPOKED+22,12:DU=0
265 IFC1=13ANDZ=9THENPOKED+22,14:GOSUB430
270 IFC1=0ANDSA>0THENPOKED+22,13:SA=SA-1:AN=AN+1:PRINTCHR$(1
9)SPC(10)SA:GOSUB440
275 IFC2=19ANDZ=7THENGOTO310
280 IFC2=11ANDSE>0ANDSA=0THENZ=10:SE=SE-(8-SA):SA=8:POKEBS+1
1,56:GOSUB430
285 IFC2=2ANDSA=0THENZ=9:GOSUB430
290 IFC2=5ANDSA=0THENZ=8:GOSUB430
295 IFC2=6ANDSA=0THENZ=7:GOSUB430
```



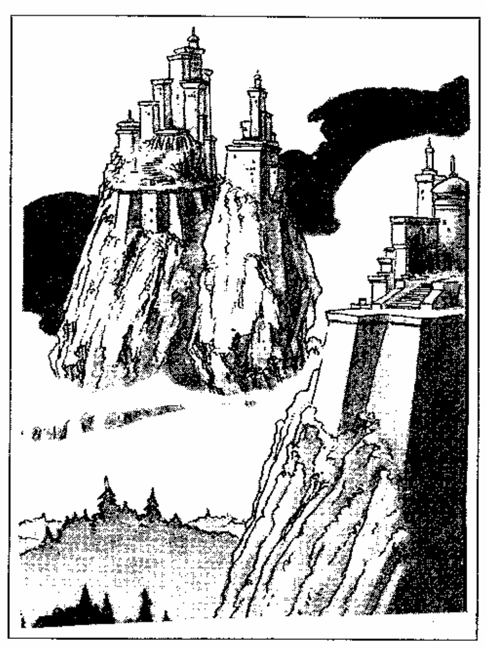

```
300 IFC2=21ANDSA=0ANDZ=9THENDU=1:GOSUB430
305 RETURN
310 FORN=0TOBL:POKEBZ-N,12:POKEFZ-N,2
315 POKET-2,230-(N*10):FORI=1TO100:NEXT
320 PRINT"V"BL-N
325 NEXT
330 BZ=BZ-BL:FZ=FZ-BL:BL=0:POKET-2,0
335 RETURN
350 PRINTCHR$(147):POKE36869,240:POKE36879,93:POKET,0
355 IFSN>SOTHENSO=SN
360 PRINTTAB(7)"DU HAST"SPC(39)""SN""SPC(38)"PFLANZEN"S
PC(36)"GEERNTET"
365 PRINT:FORN=0TO21:PRINT"±";:NEXT
370 PRINTSPC(27)"HOECHSTERNTE"SPC(37)""SO""SPC(38)"PFLANZ
EN"
375 PRINTSPC(44)" FUER EIN NEUES SPIEL DRUECKE DIE ROTE T
ASTE"
380 FORM=253TO248STEP-.04:POKET-3,M:NEXT
385 SYS828:IFPEEK(252)=0THENPOKET-3,0:SN=0:GOTO30
390 FORM=249TO252STEP.02:POKET-3,M:NEXT
395 GOTO380
400 FORN=10TO50STEP5:POKET-1,200+N:NEXT:POKET-1,0:RETURN
410 FORN=50TO0STEP-5:POKET-2,230-N:NEXTN:POKET-2,0:RETURN
430 FORN=1TO20STEP2:POKET-1,210+N:NEXT:POKET-1,0
435 PRINTCHR$(19)SPC(15)SE:IFSE<10THENPOKE7697,32
440 FORN=1TO10:POKET-1,242+N:NEXT:POKET-1,0:RETURN
450 FORN=0TO511:POKE7168+N,PEEK(32768+N):NEXT
460 FORN=0TO183:READA:POKE7168+N,A:S=S+A:NEXT
470 FORN=0TO34:READA:POKE828+N,A:S=S+A:NEXT
480 IFS<>18504THENPRINTSPC(30);"FEHLER":END
490 RETURN
500 DATA0,0,0,129,195,255,255,0,28,20,8,127,93,28,20,20
505 DATA12,18,146,158,94,62,30,30,28,42,20,42,28,8,8,8
510 DATA129,66,60,24,129,195,255,0,106,86,106,86,2,2,2,2
515 DATA17,31,127,223,159,223,127,31,56,40,17,255,191,60,40,
40
520 DATA181,85,178,23,31,7,5,5,7,133,130,79,119,119,117,5
525 DATA7,5,50,47,231,231,5,5,129,129,213,171,213,171,213,25
5
530 DATA24,66,24,165,165,24,66,24,0,0,0,153,153,195,255,0
535 DATA24,36,36,24,153,255,255,0,0,102,134,137,111,41,201,0

540 DATA0,0,0,0,0,0,0,0,0,55,50,74,122,74,74,0
545 DATA0,0,0,0,129,255,255,0,255,0,0,0,0,0,0,255
550 DATA128,64,32,16,8,4,2,1,64,160,72,9,7,7,7,7
555 DATA0,238,168,238,136,136,137,0,169,127,141,34,145,173,3
2,145
560 DATA41,128,133,251,169,255,141,34,145,173,31,145,41,28,2
4,101
565 DATA251,133,251,173,31,145,41,32,133,252,96
 READY.
```



Schloß Gruselstein

Ihre Aufgabe besteht darin, 5 Gespenster, welche in der Burg herumgeistern, einzufangen bzw. einzumauern. für die Grundversion des VC-20



Bevor Sie nun ans „Gespensterfangen" gehen, muß noch erwähnt werden, daß diese nur in einer der 16 Ruinen eingemauert werden dürfen. Damit die Sache noch erschwert wird, werden Sie von einem Monster verfolgt. Aber Achtung: Das Monster sucht den kürzesten Weg zu Ihnen und zerstört rücksichtslos alle von Ihnen errichteten Mauern. Haben Sie es geschafft, ein Gespenst einzumauern, dann verschwindet dieses, und das Spiel geht mit einem Gespenst weniger weiter. Sind alle Gespenster eingemauert, erscheinen 5 neue Plagegeister. Dies wiederholt sich solange, bis alle 3 zur Verfügung stehenden Figuren verloren wurden.

Wenn Sie das Spiel starten, müssen Sie sich einen Augenblick gedulden (Zeichenumwandlung), dann können Sie zwischen 5 verschiedenen Schwierigkeitsgraden wählen, welche das Tempo der Monster bestimmen. Schwierigkeitsstufe >1< ist jedoch nicht zu empfehlen, auf dieser Stufe können Sie sich höchstens im „Davonlaufen" üben.

Steuerung:
Links >A<, rechts >D<, aufwärts >W<, abwärts >Leertaste<.
Wenn Sie eine Mauer bauen wollen, drücken Sie kurz die F7-Taste und sofort danach die gewünschte Richtungstaste.

```
1 REM SCHLOSS GRUSELSTEIN
2 REM COPYRIGHT BY
3 REM WINKLER GABRIEL
4 REM STOECKELWEINGARTEN 75
10 READA:IFA=-1THENCLR:GOTO70
20 FORT=0TO7:READB:POKE7168+T+A*8,B:NEXT:GOTO10
30 DATA32,,,,,,,,,63,60,126,126,165,231,255,219,153
35 DATA62,66,66,126,219,255,195,126,60,61,24,24,126,90,24,60
,36,36
40 DATA59,224,224,224,224,255,255,255,255,58,7,7,7,7,255,255
,255,255
45 DATA56,255,255,255,255,7,7,7,7,57,255,255,255,255,224,224
,224,224
50 DATA60,251,251,251,,239,239,239,,-1
70 POKE36879,10:POKE36878,15:M=3
80 PRINT"   SCHLOSS GRUSELSTEIN  ":INPUT"  SPIELSTUFE: (1-5
)";S:IFS<1ORS>5THENS=0
90 AS=5:X(1)=4:X(2)=7:X(3)=9:X(4)=4:X(5)=9:Y(1)=88:Y(2)=154:
Y(3)=88:Y(4)=198:Y(5)=198
100 A$="    9 8 9 8   9 8 9 8                       ; : ;
 :    ;  :  ;  :      "
110 POKE36869,255:PRINT"      "A$;" "A$;"  "A$;" "A$:FORT=0T
O21
120 POKE38422+T,4:POKE38884+T,4:POKE38422+T*22,4:POKE38443+T
*22,4:POKE7702+T,60
130 POKE8164+T,60:POKE7702+T*22,60:POKE7723+T*22,60:NEXT:GOS
UB3010:GOTO4000
1000 P=PEEK(197):A=0:B=0
1010 IFP=17THENA=-1:GOTO1070
1020 IFP=18THENA=1:GOTO1070
1030 IFP=9THENB=-22:GOTO1070
1040 IFP=32THENB=22:GOTO1070
1050 IFP=63THEN2000
1060 GOTO1100
1070 POKE36876,200:POKE8140+X+Y,32:X=X+A:Y=Y+B:POKE36876,0
1080 IFPEEK(8140+X+Y)>61THENGOSUB3000
1090 IFPEEK(8140+X+Y)<>32THENX=X-A:Y=Y-B
1100 POKE38860+X+Y,7:POKE8140+X+Y,61:RETURN
2000 FORT=0TO50:P=PEEK(197)
2010 IFP=17THENA=-1
2020 IFP=18THENA=1
```

```
2030 IFP=9THENA=-22
2040 IFP=32THENA=22
2050 IFATHEN2070
2060 NEXT:RETURN
2070 IFPEEK(8140+X+Y+A)<>32THENRETURN
2080 FORT=150TO250STEP8:POKE36877,T:NEXT
2090 POKE38860+X+Y+A,3:POKE8140+X+Y+A,60:POKE36877,0:RETURN
3000 FORT=130TO250:POKE36876,T:NEXT:POKE36876,0:M=M-1:IFM=0T
HEN7000
3010 POKE8140+X+Y,32:POKE8140+X1+Y1,32:X=-1:Y=0:X1=-10:Y1=0
3020 PRINT"▯";:FORT=1TOM:PRINT"▯= ▮";:NEXT:PRINT"▯"TAB(9)"▯S
CORE:"SC"▀":RETURN
4000 Q=Q+1:IFQ>ASTHENQ=1
4010 MO=MO+1:IFMO=STHENMO=0:GOSUB8000
4020 POKE7794+X(Q)+Y(Q),32:R(Q)=R(Q)-1:IFR(Q)>0THEN4040
4030 A(Q)=0:B(Q)=0:R(Q)=INT(RND(1)*6):Z%=RND(1)*5:ONZ%GOSUB4
150,4160,4170,4180
4040 X(Q)=X(Q)+A(Q):Y(Q)=Y(Q)+B(Q)
4050 IFPEEK(7794+X(Q)+Y(Q))=61THENGOSUB3000
4060 IFPEEK(7794+X(Q)+Y(Q))<>32ORX(Q)<0ORX(Q)>13ORY(Q)<0ORY(
Q)>286THEN4500
4070 POKE38514+X(Q)+Y(Q),1:POKE7794+X(Q)+Y(Q),63:GOSUB4200:G
OSUB1000:GOTO4000
4150 A(Q)=1:RETURN
4160 A(Q)=-1:RETURN
4170 B(Q)=22:RETURN
4180 B(Q)=-22:RETURN
4200 L=7794+X(Q)+Y(Q):IFPEEK(L-21)<>56THENRETURN
4210 IFPEEK(L+1)<>60ORPEEK(L-1)<>60ORPEEK(L+22)<>60ORPEEK(L-
22)<>60THENRETURN
4220 FORT=130TO250:POKE36875,T:NEXT:POKE36875,0
4230 POKEL,32:POKEL+1,32:POKEL-1,32:POKEL+22,32:POKEL-22,32
4240 X(Q)=X(AS):Y(Q)=Y(AS):SC=SC+100:GOSUB3020:AS=AS-1:IFAS=
0THEN5000
4250 RETURN
4500 X(Q)=X(Q)-A(Q):Y(Q)=Y(Q)-B(Q):R(Q)=0:GOTO4070
5000 POKE36869,240:PRINT"▯▯▯DURCHGANG: ";SC/500:PRINT"PUNKT
E:",SC:PRINT"MAURER:",M
5010 PRINT"▯WOLLEN SIE DAS SPIEL  FORTSETZEN ?":PRINT"▯▯▯
BITTE EINE TASTE"
5020 GOSUB7500:GOTO90
7000 PRINT"▯ ":GOSUB7500:POKE36869,240
7010 PRINT"▯▯▯▯▯▯▯▯▯▯   NOCH EIN SPIEL ?       BITTE EINE TAS
TE"
7020 GOSUB7500:RUN70
7500 POKE198,0:WAIT198,1:RETURN
8000 POKE8140+X1+Y1,32
8020 IFX1>XTHENX1=X1-1:M1=-1
8030 IFX1<XTHENX1=X1+1:M1=1
8040 GOSUB9000:IFATHENY1=Y1+M2:GOTO8080
8050 IFY1>YTHENY1=Y1-22:M2=-22
8060 IFY1<YTHENY1=Y1+22:M2=22
8070 GOSUB9000:IFATHENX1=X1+M1
8080 IFPEEK(8140+X1+Y1)=61THENGOSUB3000
8085 IFPEEK(8140+X1+Y1)=60THENFORT=150TO200:POKE36877,T:NEXT

8090 POKE36877,200:POKE38860+X1+Y1,1:POKE8140+X1+Y1,62:POKE3
6877,0:GOSUB1000:RETURN
9000 IFPEEK(8140+X1+Y1)>59THEN9020
9010 IFPEEK(8140+X1+Y1)<>32THENA=1:RETURN
9020 A=0:RETURN
RREADY.
```



Topprogramm

Ein kleiner Floh und sein Freund, lebenslustig und froh, haben ein schweres Schicksal zu meistern. Sie sind beim Herumhüpfen unbewußt in den Einflußbereich eines Magiers geraten.

Zur Strafe sollen sie im Kellergewölbe der Behausung des Magiers eingesperrt werden. Doch siehe da: Nur bei einem Floh klappt der Zauber. Immerhin: Auch der zweite ist im Bann des Magiers, denn er befindet sich im Erdgeschoß des Zauberreiches, ohne einen sichtbaren Weg in die Freiheit.
Nun gilt es für diesen kleinen Floh, für sich und seinen Freund einen Ausweg zu finden.
Sein Weg durch das Zauberreich (6 Bilder), über Rolltreppen (nicht immer sichtbar), automatische Kletterstangen, fliegende Teppiche und Sprungfedern endet im Kellergewölbe, nachdem hier drei tickende Bomben ins Wasser geworfen, der zweite Floh befreit und der Ausgang gefunden ist.
Den Weg erschweren Teufelsmonster, fliegende Vögel und rollende Kugelmonster.
Aufgabe des Spielers ist es, den Floh von einer zur nächsten Plattform hüpfend, ins nächste Bild zu bringen.
Um in den ersten zwei Bildern voranzukommen, müssen unter der Decke hängende Hebel bedient werden.
Am Spielende erscheint eine „High-Score"-Tabelle.

Prinzip der Punktezählung:
Für jedes Bild gibt es eine Vorgabe von 800 Zeiteinheiten.
Während des Spieldurchlaufs wird rückwärts gezählt und die verbleibende Restsumme wird als Punkte gutgeschrieben. Im Schlußbild (Keller) gilt es in 120 Zeiteinheiten die 3 Bomben zu „entschärfen". Die Differenz wird als Punkte gutgeschrieben. Außerdem gibt es bei Befreiung des zweiten Flohs als Bonus 400 Punkte.
Als Reserve stehen fünf weitere Spielfiguren zur Verfügung.
Der Verlust jeden „Lebens" wird mit 100 Minuspunkten bestraft.
Wenn der Floh in ein unteres Bild (Stockwerk) zurückfällt, werden 800 Punkte abgezogen. Bei Wiedererreichen des verlassenen Stockwerkes werden diese 800 Punkte, abzüglich verstrichene Zeiteinheiten, wieder zugezählt, so daß ein Absturz nur mit dem entsprechenden Zeitverlust bestraft wird.
Bei extrem langsamen Spiel können sich aber auch Minuszahlen in der Punktebewertung ergeben. In diesem Fall gibt es natürlich keine „High-Score"-Bewertung.

Zum Programm
Ein umfangreiches Basic-Programm wie das vorliegende, würde – müßten stets alle für den gesamten Spielablauf notwendiges Abfragen durchlaufen werden – für den Benutzer sehr ermüdend sein.
Deshalb mußte bei der Programmerstellung systematisch vorgegangen werden.
Die Hauptroutine (ab Zeile 5000) enthält nur alle für den Gesamtspielablauf unbedingt erforderlichen Befehle.
Alle Aufgaben, die nur einen speziellen Spielabschnitt betreffen, sind in Unterroutinen geordnet, zusammengefaßt und werden aus Zeile 5032 der Hauptroutine (gosub g) angesprungen. (Die Variable g wird in den Zeilen 7000–7500 festgelegt.)
Diese Unterroutinen wurden wegen des schnelleren Zugriffs direkt an den Programmanfang gestellt (Zeilen 19 bis 400).
Nach der Hauptroutine folgen die Unterprogramme für den jeweiligen Bildaufbau.
Alles weitere, wie Anfangsvariablenbestimmung, UDG's usw. steht am Programmende.
Es wurden möglichst viele Befehle in eine Zeile geschrieben und es wurde ein logische Schreibweise bevorzugt wie zum Beispiel: IF NOT i THEN ... statt IF i = 0 THEN ...
Auf eine laufende Punkteanzeige wurde wegen des Zeitbedarfs verzichtet.
Dafür gibt es eine stete Punkteanzeige (aus grafischen Gründen im unteren Bildschirmbereich), die nach jedem Bildwechsel auf den aktuellen Zwischenstand gebracht wird.
Nach unseren Erfahrungen erhöht dies sogar die Spannung.
Die Variablen-Bestimmung findet man sehr übersichtlich im Listing ab Zeile 9100.
Sämtliche selbstdefinierten UDG's sind als CHR (X)-Werte im Programm, um die Eingabe etwas zu erleichtern.

```
   1 REM *** JUMP ABOUT ***

   © by Heinz-Guenther Grebe
        Schlangenstr. 15

   2
   3 LET m6=0: LET m7=0: LET m8=
0: LET c$="-": LET d$="-": LET e
$="-": REM VARIABLEN FUER HIGH-S
CORE
   4 POKE 23609,17: REM TASTATUR
-PIEP
   5 CLS : INK 0: PAPER 7: GO SU
B 9200: GO TO 9500
   9 INK 0: PAPER 7: GO SUB 9100
: REM VARIABLEN
  10 GO SUB 9000: GO SUB 7000: G
O SUB 20: GO TO 5000
  19 REM HEBEL
  20 IF p=2 THEN GO TO 110
  21 IF p>2 THEN RETURN
  22 INK 4: PRINT AT 2,3;CHR$ 15
1;AT 2,27;CHR$ 151;AT 6,3;CHR$ 1
51;AT 10,5;CHR$ 151;AT 10,28;CHR
$ 151;AT 14,4;CHR$ 151;AT 18,10;
CHR$ 151: INK 0: RETURN
  29 REM ROLLTREPPE
  30 IF p=2 THEN GO TO 100
  32 IF e=20 THEN LET r=24: GO T
O 40
  33 IF e=16 THEN LET r=15: GO T
O 40
  34 IF e=12 AND w=5 THEN LET r=
22: GO TO 40
  35 IF e=12 AND w=28 THEN LET r
=13: GO TO 42
  36 IF e=8 AND w=3 THEN LET r=1
6: GO TO 40
  37 IF e=4 AND w=3 THEN PRINT A
T 2,23;" ": BEEP .01,11: PRINT A
T 3,23;" ": BEEP .01,13: PRINT A
T 4,23;" ": BEEP .01,15: RETURN
  38 IF e=4 AND w=27 THEN LET r=
8: GO TO 42
  40 INK 3: PRINT AT e,r;CHR$ 14
4;CHR$ 145;AT e-1,r+1;CHR$ 144;C
HR$ 145;AT e-2,r+2;CHR$ 144;CHR$
 145;AT e-3,r+3;CHR$ 144: INK 0:
 RETURN
  42 INK 3: PRINT AT e,r-1;CHR$
146;CHR$ 147;AT e-1,r-2;CHR$ 146
;CHR$ 147;AT e-2,r-3;CHR$ 146;CH
R$ 147;AT e-3,r-3;CHR$ 147: INK
```




```
8: RETURN
  49 REM ROLLTREPPE WEG
  51 PRINT AT e,w;" "
  52 IF q=1 THEN LET w=w+1
  53 IF q=2 THEN LET w=w-1
  54 PRINT AT e,w; INK 1;CHR$ 14
8
  55 IF p>1 THEN RETURN
  56 PRINT AT 1,5;" ";AT 2,5;"
";AT 3,6;"  ";AT 4,7;"  ";AT 5,1
9;" ";AT 6,18;"  ";AT 7,17;"  ";
AT 8,16;"  ";AT 9,10;" ";AT 10,1
0;"  ";AT 11,11;"  ";AT 12,12;"
";AT 9,25;" ";AT 10,24;"  ";AT
11,23;"  ";AT 12,22;"  ";AT 13,1
8;" ";AT 14,17;"  ";AT 15,16;"
";AT 16,15;"  ";AT 17,27;" ";AT
18,26;"  ";AT 19,25;"  ";AT 20,2
4;"
  57 GO SUB 20
  58 RETURN
  60 REM TEUFEL BILD 1
  61 IF e<5 THEN PRINT AT x,8;"
": GO TO 73
  62 PRINT AT y,26;" ": PRINT AT
 x,8;" ": LET x=x+1: PRINT AT x,
8; PAPER 6; BRIGHT 1;CHR$ 159: I
F x>15 THEN LET x=6
  63 IF ATTR (e,w)=112 THEN GO S
UB 8000: PRINT AT e,w;" ": LET e
=20: LET w=2
  64 PRINT AT 16,8;" "
  65 RETURN
  72 REM 2.TEUFEL BILD 1
  73 PRINT AT y,26;" ": LET y=y+
1: PRINT AT y,26; PAPER 6; BRIGH
T 1;CHR$ 159: IF y>7 THEN LET y=
2
  74 IF ATTR (e,w)=112 THEN GO S
UB 8000: PRINT AT e,w;" ": LET e
=20: LET w=2
  75 PRINT AT 8,26;" "
  76 RETURN
  80 REM ZAEHLEN DER SPRUENGE
         VOM FLOH
  82 IF v>4 THEN LET v=0: GO TO
88
  86 RETURN
  88 IF ATTR (e+1,w)=56 THEN PRI
NT AT e,w;" ": LET e=e+1: PRINT
AT e,w; INK 1;CHR$ 148: BEEP .01
,11: GO TO 88
  90 RETURN
 100 REM HEBELABFRAGE 2.BILD
 104 IF e=8 THEN FOR i=2 TO 4: P
RINT AT i,15;" ": BEEP .01,i: NE
XT i: RETURN
 106 IF e=12 THEN FOR i=10 TO 12
: PRINT AT i,15;" ": BEEP .01,1:
 NEXT i: RETURN
 110 REM HEBEL 2.BILD
 112 INK 4: PRINT AT 6,27;CHR$ 1
51;AT 10,27;CHR$ 151: INK 0: RET
URN
 130 REM MONSTER-KUGEL 3.BILD
 131 LET z=z+1
 136 IF l=1 THEN PRINT AT k,l;"
": LET k=2: LET l=27
 137 IF l=18 THEN PRINT AT k,l;"
": LET k=2: LET l=17
 138 PRINT AT k,l;" "
 140 IF NOT ATTR (k+1,l)=61 THEN
 LET k=k+1: GO TO 146
 142 LET l=l-1
 146 IF ATTR (k,l)=57 THEN GO SU
B 8000: PRINT AT e,w;" ": LET e=
20: LET w=12
 148 IF NOT t THEN PRINT AT k,l;
 INK 2;CHR$ 152: LET t=1: RETURN

 149 PRINT AT k,l; INK 2;CHR$ 16
0: LET t=0: RETURN
 160 REM VOGEL 3.BILD
 162 IF ATTR (e+1,w)=62 THEN PRI
NT AT e,w;CHR$ 163;AT e-1,w; INK
 1;CHR$ 148: BEEP .01,31: BEEP .
01,33: BEEP .01,31: BEEP .01,33:
 FOR i=1 TO 5: PRINT AT e,w;" ":
 LET e=e-1: PRINT AT e,w; INK 1;
CHR$ 148: BEEP .01,23: NEXT i: P
RINT AT e,w;" ": LET w=w+1: PRIN
T AT e,w; INK 1;CHR$ 148;AT e,w;
" "
 164 IF NOT w THEN CLS : GO SUB
7300: GO SUB 8009: RETURN
 166 PRINT AT 12,o;" ": LET o=o+
1: IF o>30 THEN LET o=1
 167 IF ATTR (12,o)=57 THEN GO S
UB 8000: PRINT AT e,w;" ": LET e
=20: LET w=2
 168 IF NOT t THEN PRINT AT 12,o
; INK 2;CHR$ 161: LET t=1: RETUR
N
 169 PRINT AT 12,o; INK 2;CHR$ 1
62: LET t=0: RETURN
 170 REM FLIEGENDER TEPPICH UND
         VOGEL 4.BILD
 171 IF e=21 THEN CLS : GO SUB 7
400: GO SUB 8009: RETURN
 172 PRINT AT 9,o; PAPER 7; INK
0;"  ": PRINT AT 8,o+1; PAPER 7;
 INK 0;" "
 173 LET o=o-1: IF o<2 THEN LET
o=29
 174 IF NOT t THEN PRINT AT 9,o;
 INK 4;CHR$ 154;CHR$ 155: LET t=
1: GO TO 176
 175 PRINT AT 9,o; INK 4;CHR$ 15
5;CHR$ 154: LET t=0
 176 IF e=8 AND ATTR (e+1,w)=60
THEN PRINT AT e,w;" ": LET w=0:
PRINT AT e,w; INK 1;CHR$ 148
 179 IF ATTR (8,c)=57 THEN GO SU
B 8000: LET e=4: LET w=28
 180 PRINT AT 8,c;" ";: LET c=c+
1: IF c>30 THEN LET c=1
 181 IF NOT t THEN PRINT AT 8,c;
 INK 2;CHR$ 161: GO TO 183
 182 PRINT AT 8,c; INK 2;CHR$ 16
2
 183 IF w<7 THEN GO TO 188
 184 IF e>15 THEN PRINT AT e,w;"
": GO SUB 8005: LET e=4: LET w=
27
 186 REM BERG OEFFNEN 4.BILD
 188 IF w=2 AND e<12 THEN PRINT
AT 12,2;" ";AT 13,2;" ";AT 14,2;
" ";AT 15,1;"  ";AT 16,1;"  ";
AT 17,1;" ";AT 18,1;" ";AT 19,1;
"     ";AT 20,1;"     ";AT 21,
4;"  "
 189 RETURN
 190 REM FLIEGENDER TEPPICH
         VORGARTEN 5.BILD
 191 IF ATTR (e,w-1)=249 THEN PR
INT AT 2,21;"  ": FOR i=1 TO 20:
 PRINT AT i,27;"   ": BEEP .01,
i: NEXT i: FOR i=18 TO 20: PRINT
 AT i,31;" ": NEXT i: LET w=21
 192 IF w>30 THEN LET z1=z1+(800
-z): GO SUB 7000: FOR i=18 TO 20
: PRINT AT i,0; INK 2;"█": NEXT
i: PRINT AT 21,5;"   ": LET e=20
: LET w=1: LET p=1: RETURN
 193 IF ATTR (e+1,w)=62 THEN PRI
NT AT e,w;CHR$ 163;AT e-1,w; INK
 1;CHR$ 148: BEEP .01,31: BEEP .
01,33: BEEP .01,31: BEEP .01,33:
 FOR i=1 TO 5: PRINT AT e,w;" ":
 LET e=e-1: PRINT AT e,w; INK 1;
CHR$ 148: BEEP .01,23: NEXT i: P
RINT AT e,w;" ": LET w=w+1: PRIN
T AT e,w; INK 1;CHR$ 148;AT e,w;
" "
 194 PRINT AT 14,d; PAPER 7; INK
 0;"  ": PRINT AT 13,d-1; PAPER
7; INK 0;" "
 196 LET d=d+1: IF d>23 THEN LET
d=8: PRINT AT 13,20;"   "
 198 IF NOT t THEN PRINT AT 14,d
```

```
; INK 2;CHR$ 154;CHR$ 155: LET t
=1: GO TO 202
 200 PRINT AT 14,d; INK 2;CHR$ 1
55;CHR$ 154: LET t=0
 202 IF e=13 AND ATTR (e+1,w)=58
 THEN PRINT AT e,w;" ": LET w=w+
1: PRINT AT e,w; INK 1;CHR$ 148
 204 IF ATTR (e+1,w)=189 THEN PR
INT AT e,w;" ": GO SUB 8005: LET
 e=3: LET w=5
 206 RETURN
 220 REM BOMBEN + TEUFEL 6.BILD
 221 IF n>3 THEN GO TO 228
 222 IF n=3 THEN FOR i=15 TO 18:
 PRINT AT i,13;" ": NEXT i: LET
n=4: GO TO 228
 223 PRINT AT 2,24;z: IF z=120 T
HEN PRINT AT 2,11; FLASH 1; INK
2;"LEIDER NICHT";AT 3,11; FLASH
1; INK 2;"GESCHAFFT!!!": FOR i=1
 TO 30: FOR j=1 TO 6: BORDER j:
BEEP .01,j: NEXT j: NEXT i: BORD
ER 5: LET i=0: LET ii=0
 228 PRINT AT x,22;" ": LET x=x+
1: PRINT AT x,22; PAPER 6; BRIGH
T 1;CHR$ 159: IF x>10 THEN LET x
=4
 230 IF ATTR (e,w)=112 THEN GO S
UB 8000: LET e=7: LET w=9
 231 IF x=4 THEN PRINT AT 11,22;
" "
 233 IF w=3 THEN PRINT AT 9,0;"
 ";AT 10,0;"   ": PRINT AT 10,1
; INK 1;CHR$ 148: LET i=0
 235 IF ATTR (e,w+1)=184 THEN GO
 TO 243
 237 IF ATTR (e+1,w)=189 THEN PR
INT AT e,w;" ": GO SUB 8000: LET
 e=4: LET w=5: RETURN
 240 RETURN
 243 IF e=7 THEN PRINT AT 6,28;"
 ";AT 7,27;" ": INK 1;CHR$ 148:
LET w=28: FOR i=6 TO 17: PRINT A
T i-1,29;" ";AT i,29;" ";AT i,29
;CHR$ 153;AT i+1,29;CHR$ 152: BE
EP .01,13: NEXT i: PRINT AT 17,2
9;" ";AT 18,29;" ": LET n=n+1
 245 IF e=11 THEN PRINT AT 10,27
;" ";AT 11,26;" ": INK 1;CHR$ 14
8: LET w=27: FOR i=10 TO 17: PRI
NT AT i-1,28;" ";AT i,28;" ";AT
i,28;CHR$ 153;AT i+1,28;CHR$ 152
: BEEP .01,13: NEXT i: PRINT AT
17,28;" ";AT 18,28;" ": LET n=n+
1
 247 IF e=15 THEN PRINT AT 14,26
;" ";AT 15,25;" "; INK 1;CHR$ 14
8: LET w=26: FOR i=14 TO 17: PRI
NT AT i-1,27;" ";AT i,27;" ";AT
i,27;CHR$ 153;AT i+1,27;CHR$ 152
: BEEP .01,13: NEXT i: PRINT AT
17,27;" ";AT 18,27;" ": LET n=n+
1
 250 RETURN
 400 REM ABFRAGE WELCHES BILD
 402 IF NOT p THEN LET b=7500
 404 IF p=1 THEN LET b=7000
 406 IF p=2 THEN LET b=7110
 408 IF p=3 THEN LET b=7210
 420 RETURN
5000 REM HAUPTROUTINE
5005 LET f=0: REM HEBEL GETROFFE
N?
5006 LET q=0: REM AUF TREPPE SCH
RITT WEITER
5010 IF INKEY$="n" THEN PRINT AT
 e,w;" ": IF ATTR (e,w-1)=56 THE
N LET w=w-1
5012 IF INKEY$="m" THEN PRINT AT
 e,w;" ": IF ATTR (e,w+1)=56 THE
N LET w=w+1
5013 IF w>31 THEN LET w=1: LET p
=p-1: LET ru=0: CLS : GO SUB 721
0: GO SUB 8009: LET e=4
5014 IF ATTR (e+1,w)=56 THEN PRI
NT AT e,w;" ": LET e=e+1
5016 IF INKEY$="N" THEN PRINT AT
 e,w;" ": IF ATTR (e-1,w-1)=56 T
HEN LET e=e-1: LET w=w-1: LET v=
v+1: GO SUB 80
5018 IF INKEY$="M" THEN PRINT AT
 e,w;" ": IF ATTR (e-1,w+1)=56 T
HEN LET e=e-1: LET w=w+1: LET v=
v+1: GO SUB 80
5019 IF ATTR (e-2,w)=60 THEN LET
 f=1
5020 IF NOT ATTR (e+1,w)=56 THEN
 LET v=0
5021 IF w>31 THEN LET w=1: LET p
=p-1: LET ru=0: CLS : GO SUB 723
0: GO SUB 8009: LET e=4
5022 IF INKEY$="z" THEN FOR i=1
TO 2: PRINT AT e,w;" ": LET e=e-
1: PRINT AT e,w; INK 1;CHR$ 148:
 BEEP .02,19: NEXT i: PRINT AT e
,w;" ": LET e=e+2: IF f=1 THEN G
O SUB 30
5030 IF NOT a THEN PRINT AT e,w;
 INK 1;CHR$ 148: LET a=1: GO TO
5032
5031 PRINT AT e,w; INK 1;CHR$ 14
9: LET a=0
5032 LET z=z+1: GO SUB g: IF NOT
 i THEN GO TO 9900
5033 IF ATTR (e,w+1)=59 THEN PRI
NT AT e,w;" ": LET e=e-1: LET w=
w+1: LET q=1: BEEP .02,21: GO TO
 5030
5034 IF ATTR (e,w-1)=59 THEN PRI
NT AT e,w;" ": LET e=e-1: LET w=
w-1: LET q=2: BEEP .02,21: GO TO
 5030
5040 IF ATTR (e,w+1)=51 THEN PRI
NT AT e,w;" ": LET e=e-1: LET q=
1: BEEP .01,17: GO TO 5030
5044 IF ATTR (e,w-1)=51 THEN PRI
NT AT e,w;" ": LET e=e-1: LET q=
2: BEEP .01,17: GO TO 5030
5046 IF q<>0 THEN GO SUB 50: GO
SUB 20
5058 IF NOT e AND w=4 THEN LET p
=p+1: GO SUB 400: LET e=20: LET
w=3: CLS : GO SUB b: GO SUB 8009
5060 IF e=21 THEN LET ru=0: LET
p=p-1: CLS : GO SUB 400: GO SUB
b: GO SUB 8009: LET e=0: LET w=6
: GO SUB 20
5100 GO TO 5000
7000 REM BILDAUFBAU ERDGESCHOSS
     (1.BILD)
7001 LET z=0: IF NOT ru THEN LET
 z1=z1-800: LET ru=1
7005 CLS : LET g=60
7008 INK 5
7010 PRINT AT 1,1;"███████";AT 1,8;
"████████████████████████";AT 5,2
;"███";AT 5,7;"████████████";AT 5,2
0;"██████";AT 5,27;"███"
7012 PRINT AT 9,3;"████";AT 9,9;"
█";AT 9,13;"██████";AT 9,21;"███";
AT 9,26;"█████";AT 13,4;"█████";AT
13,10;"███████";AT 13,19;"████████";
AT 13,27;"███";AT 17,3;"████";AT 1
7,8;"████";AT 17,14;"█████";AT 17,
21;"████";AT 17,28;"███"
7013 FLASH 1: PRINT AT 2,23;"█";
AT 3,23;"█";AT 4,23;"█": FLASH 0
7014 INK 2
7016 FOR i=0 TO 21: PRINT AT i,0
;"█";AT i,31;"█": NEXT i
7018 PRINT AT 21,1;"██████████████
████████████████"
7019 PRINT AT 0,9;"ERDGESCHOSS"
7030 INK 0: GO SUB 8009: GO SUB
7180: RETURN
7100 REM BILDAUFBAU 1.ETAGE
     (2.BILD)
7105 LET e=20: LET w=4
7110 CLS : LET z1=z1+(800-z): LE
T z=0: IF NOT ru THEN LET z1=z1-
```

```
1600: LET ru=1
7120 INK 5: PRINT AT 1,1;"▀▀▀▀";AT 1,6;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀";AT 5,6;"▀▀▀▀▀▀▀▀▀▀▀";AT 5,20;"▀▀▀▀";AT 5,27;"█";AT 5,30;"▀▀";AT 9,2;"▀▀";AT 9,6;"▀▀▀▀▀";AT 9,13;"▀▀▀▀▀";AT 9,26;"██";AT 13,10;"▀▀▀▀▀▀▀";AT 13,19;"▀▀";AT 13,23;"▀▀▀▀▀";AT 17,5;"▀▀▀▀▀▀▀▀";AT 17,22;"▀▀▀";AT 17,29;"▀▀";AT 21,1;"▀▀▀▀▀";AT 21,8;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"
7122 INK 2: FOR i=0 TO 21: PRINT AT i,0;"█";AT i,31;"█": NEXT i
7124 LET u$=CHR$ 164: INK 3: PAPER 6: PRINT AT 1,4;u$;AT 2,4;u$;AT 3,4;u$;AT 5,19;u$;AT 6,19;u$;AT 7,19;u$;AT 8,19;u$;AT 5,29;u$;AT 6,29;u$;AT 7,29;u$;AT 8,29;u$;AT 13,20;u$;AT 14,20;u$;AT 15,20;u$;AT 17,28;u$;AT 18,28;u$;AT 19,28;u$;AT 20,28;u$: FOR j=9 TO 15: PRINT AT j,3;u$: NEXT j
7126 PAPER 7
7130 INK 5: FLASH 1: PRINT AT 2,15;"▚";AT 3,15;"▚";AT 4,15;"▚";AT 10,15;"▞";AT 11,15;"▞";AT 12,15;"▞": FLASH 0
7134 INK 4: PRINT AT 6,27;CHR$ 151;AT 10,27;CHR$ 151: INK 0
7136 PRINT AT 20,14; INK 5;"█";AT 20,15; INK 5; FLASH 1;"▞▚": FLASH 0: INK 0
7137 PRINT AT 0,9;"1. ETAGE"
7140 LET p=2: LET g=130
7145 GO SUB 7180
7150 RETURN
7179 REM SCORE
7180 PRINT #0;"              "; PAPER 2; INK 7;" S C O R E : ";z1;"  ": RETURN
7200 REM BILDAUFBAU DACHGESCHOSS (3.BILD)
7205 CLS : LET e=20: LET w=4
7210 LET z1=z1+(800-z): LET z=0: IF NOT ru THEN LET z1=z1-1600: LET ru=1
7211 PAPER 2: INK 7: FOR i=0 TO 31: PRINT AT 1,i;CHR$ 156;AT 2,i;CHR$ 157: NEXT i: INK 2: PAPER 7: PRINT AT 3,31;"█";AT 4,31;"█": FOR i=5 TO 21: PRINT AT i,0;"█";AT i,31;"█": NEXT i
7212 INK 5: PRINT AT 5,1;"▀▀";AT 5,9;"▀▀▀▀";AT 5,19;"█▀▀";AT 5,25;"▀▀";AT 6,18;"█▀ ";AT 6,30;"▀▀";AT 7,18;"█ ";AT 8,18;"█ ";AT 9,12;"▀▀";AT 9,18;"█ ";AT 9,20;"▀▀▀▀▀";AT 8,6;"▀▀▀▀";AT 10,17;"█▀ ";AT 10,27;"▀▀▀";AT 11,16;"█▀ ";AT 13,3;"▀▀▀▀▀▀";AT 13,16;"█ ";AT 13,20;"▀▀▀▀▀▀▀▀";AT 13,30;"▀▀";AT 14,16;"█ ";AT 15,16;"█ ";AT 16,16;"█ ";AT 17,1;"▀▀";AT 17,3;"▀▀▀▀▀▀▀";AT 17,16;"█▀▀";AT 17,23;"▀▀▀▀▀▀";AT 18,10;"▀▀";AT 18,15;"▀▀ ";AT 21,1;"▀▀▀▀▀▀▀";AT 21,8;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀";AT 21,23;"▀▀▀▀▀▀▀▀"
7214 INK 6: PRINT AT 10,29;"▀▀";AT 13,11;"▀▀";AT 13,19;"▀▀";AT 17,2;"▀▀";AT 21,22;"▀▀"
7216 INK 3: PAPER 6: PRINT AT 5,2;u$;AT 6,2;u$;AT 7,2;u$;AT 8,2;u$;AT 9,2;u$: PAPER 7: PRINT AT 13,29;CHR$ 144;AT 14,28;CHR$ 144;CHR$ 145;AT 15,27;CHR$ 144;CHR$ 145;AT 16,26;CHR$ 144;CHR$ 145: INK 0
7218 INK 4: FLASH 1: FOR i=3 TO 11: PRINT AT i,14;"▚": NEXT i: FLASH 0: INK 0
7220 PRINT AT 0,9;"DACHGESCHOSS"
7230 LET p=3: LET g=160
7235 GO SUB 7180
7240 RETURN
7300 REM BILDAUFBAU BALKON (4.BILD)
7305 CLS : LET e=4: LET w=31
7310 LET z1=z1+(800-z): LET z=0: IF NOT ru THEN LET z1=z1-1600: LET ru=1
7311 INK 7: PAPER 2: FOR i=29 TO 31: PRINT AT 0,i;CHR$ 156: NEXT i: PRINT AT 1,28;CHR$ 156;CHR$ 157;CHR$ 157;CHR$ 157: PAPER 7: INK 0
7312 PRINT AT 5,26; INK 5;"▀▀▀▀▀";AT 12,0;"▀▀▀▀▀": PRINT AT 5,31; PAPER 2; INK 5;"▀▀": FOR i=6 TO 21: PRINT AT i,31; INK 2;"█": NEXT i
7313 PRINT AT 0,9;"BALKON"
7314 FOR j=12 TO 21: PRINT AT j,0; INK 5;"████": NEXT i: FOR i=19 TO 21: PRINT AT i,4; INK 5;"██": NEXT i: FOR i=7 TO 39: INK 5: PLOT i,75: DRAW 16,-35: DRAW 0,-12: DRAW 32,-28: NEXT i
7316 PRINT AT 17,7; INK 5; FLASH 1;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚";AT 18,8;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚";AT 19,9;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚";AT 20,10;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚";AT 21,11;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚"
7318 FOR i=7 TO 11: PRINT AT i,0; INK 7; PAPER 1;"█": NEXT i
7320 INK 0: PAPER 7: LET p=4: LET g=170
7325 LET o=29
7330 GO SUB 7180
7360 RETURN
7400 REM BILDAUFBAU VORGARTEN (5.BILD)
7401 LET z1=z1+(800-z)
7402 LET z=0
7405 LET e=0: LET w=5: CLS : LET p=5: LET g=190
7406 PRINT AT 0,9;"VORGARTEN"
7410 INK 5: PRINT AT 4,26;"▀▀";AT 4,4;"▀▀▀▀";AT 7,2;"▀▀▀";AT 9,21;"▀▀";AT 9,26;"▀▀";AT 12,6;"▀▀▀▀▀▀▀▀";AT 12,1;"▀▀▀";AT 13,26;"▀▀"
7411 INK 4: PRINT AT 1,9;"oooooooooooooooooo": FOR i=0 TO 11: PRINT AT i,0;"o": NEXT i: PRINT AT 13,0;"████";AT 14,0;"█████";AT 15,0;"██████";AT 16,0;"████████";AT 17,0;"█████████";AT 18,0;"██████████";AT 20,0;"████████████";AT 19,0;"███████████";AT 21,0;"█████████████";AT 19,26;"▀ ";AT 20,26;"▀ ";AT 21,26;"████";AT 17,26;"▀▀": INK 0
7413 INK 6: PRINT AT 8,8;"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀";AT 9,22;"▀▀";AT 13,25;"▀▀"
7414 INK 3: PRINT AT 5,24;"ä";AT 4,25;"ä": LET f=2: FOR i=12 TO 21: LET f=f+1: PRINT AT i,f; INK 3;"▚": NEXT i
7416 FOR i=0 TO 21: PRINT AT i,31; INK 2;"█": NEXT i
7420 INK 5: FLASH 1: PRINT AT 20,12;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚";AT 21,13;"▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚": FLASH 0
7421 INK 2: FOR i=1 TO 17: PRINT AT i,27; INK 2;"▀ ": NEXT i
7422 PLOT 216,167: DRAW 0,-128: INK 0: DRAW 31,-30
7424 PLOT 224,32: DRAW 0,48: DRAW 11,6,-2: DRAW 8,-23,-1: DRAW 0,-50
7426 PLOT 224,71: DRAW 18,-16: PLOT 224,68: DRAW 18,-16: PLOT 224,47: DRAW 18,-16: PLOT 224,44: DRAW 18,-16
7428 PLOT 227,64: DRAW 16,-35
7429 PRINT AT 18,26; INK 5;" "
7430 PRINT AT 2,21; INK 1; BRIGHT 1; FLASH 1;CHR$ 158
```



```
7433 GO SUB 7180
7434 RETURN
7500 REM BILDAUFBAU KELLER
         (6.BILD)
7502 LET z=0: LET x=4
7503 CLS
7505 PAPER 7
7510 INK 1: PAPER 6: PRINT AT 0,
0;"█████";AT 0,9;"████       ███
███████";AT 1,0;"████     ███";AT 1,3
;"██████████          ";AT 2,
0;"█████";AT 2,10;"████████████████
███████";AT 3,1;"███        ";AT 3,12;
"████████████████";AT 4,2;"█
██";AT 5,3;"███";AT 6,4;"███";AT
 7,5;"███";AT 7,18;"█";AT 8,6;"█
████████";AT 8,16;"███";AT 8,23;"█
█████"
7512 PRINT AT 9,16;"███";AT 11,0
;"████";AT 12,0;"█████";AT 12,19
;"█████████";AT 13,0;"████████";AT
 14,0;"████████";AT 15,0;"████
█";AT 16,0;"██████";AT 16,17;
"██████████";AT 17,0;"█████████████
";AT 17,16;"██████████";AT 18,0;
"████████████";AT 18,15;"███████
██"
7514 FOR i=19 TO 21: PRINT AT i,
0;"████████████████████████████████":
NEXT i: FOR i=2 TO 21: PRINT AT
i,31;"█": NEXT i
7515 FOR i=15 TO 18: PRINT AT i,
13;"█": NEXT i
7516 INK 0: PAPER 7: PRINT AT 0,
13;"KELLER"
7520 INK 3: LET j=3: FOR i=11 TO
 18: LET j=j+1: PRINT AT i,j;CHR
$ 146;CHR$ 147: NEXT i
7522 FOR i=8 TO 14: PRINT AT i,1
3; INK 3; PAPER 6;u$: NEXT i
7524 INK 7: FOR i=11 TO 14: PRIN
T AT i,12;"█": NEXT i
7530 INK 5: PAPER 7: FLASH 1: FO
R i=19 TO 21: PRINT AT i,26;"▚▚▚
▚▚": NEXT i: FLASH 0
7532 INK 0: FLASH 1: PRINT AT 6,
28;CHR$ 153;AT 7,28;CHR$ 152;AT
10,27;CHR$ 153;AT 11,27;CHR$ 152
;AT 14,26;CHR$ 153;AT 15,26;CHR$
 152: FLASH 0
7534 INK 1: PRINT AT 10,1;CHR$ 1
48: FOR i=3 TO 23 STEP 3: PLOT i
,103: DRAW 0,-15: NEXT i: PLOT 2
,103: DRAW 20,0
7540 INK 0: PAPER 7: GO SUB 8009
: LET e=0: LET w=5: LET g=220: L
ET p=6
7543 GO SUB 7180: PRINT #1;"
   BEI 120 KRACHT'S !"
7544 RETURN
7550 STOP
8000 REM FESTSTELLEN WIEVIEL
         SPIELFIGUREN
         VORHANDEN SIND
8005 LET mv=mv+1: LET z1=z1-100
8009 REM IF NOT mv THEN LET m$="
████": GO TO 8040
8010 IF mv=1 THEN LET m$="████ "
: GO TO 8040
8012 IF mv=2 THEN LET m$="███  "
: GO TO 8040
8014 IF mv=3 THEN LET m$="██   "
: GO TO 8040
8016 IF mv=4 THEN LET m$="█    "
: GO TO 8040
8018 IF mv=5 THEN LET m$="     "
: GO TO 8040
8020 IF mv=6 THEN CLS : PRINT AT
 3,7;"ALLE SPIEL-FIGUREN";AT 5,1
4;"SIND";AT 7,12;"VERLOREN!": FO
R i=40 TO 1 STEP -1: BEEP .05,i:
 NEXT i: GO TO 8050
8040 PRINT AT 0,24; INK 1;m$; IN
K 0: FOR i=1 TO 10: BEEP .01,i:
NEXT i: RETURN
```

```
8050 LET i=0: LET ii=0: RETURN
9000 REM UDG'S POKEN
9005 RESTORE
9010 FOR i=1 TO 17: READ p$
9012 FOR j=0 TO 7: READ a: POKE
USR p$+j,a: NEXT j: NEXT i
9016 DATA "h",24,24,36,90,90,36,
24,24: REM HEBEL(CHR$ 151)
9018 DATA "a",0,2,6,14,30,62,126
,254: REM ROLLTREPPE(CHR$ 144)
9020 DATA "b",0,254,252,248,240,
224,192,128: REM ROLLTREPPE(CHR$
 145)
9022 DATA "c",0,127,63,31,15,7,3
,1: REM ROLLTREPPE(CHR$ 146)
9024 DATA "d",0,64,96,112,120,12
4,126,127: REM ROLLTREPPE(CHR$ 1
47)
9026 DATA "p",36,126,255,219,255
,102,60,60: REM TEUFEL(CHR$ 159)
9028 DATA "u",24,60,126,255,255,
255,60,60: REM KLETTERSTANGE(CHR
$ 164)
9030 DATA "q",60,110,255,63,15,3
9,126,60: REM KUGELMONSTER(CHR$
160)
9033 DATA "i",60,110,255,255,255
,255,126,60: REM KUGELMONSTER/BO
MBE(CHR$ 152)
9036 DATA "t",0,126,129,126,129,
126,129,126: REM SPRUNGFEDER(CHR
$ 163)
9038 DATA "r",48,56,60,255,255,6
2,0,0: REM VOGEL(CHR$ 161)
9040 DATA "s",0,0,0,255,255,60,5
6,48: REM VOGEL(CHR$ 162)
9042 DATA "l",0,30,255,225,0,0,0
,0: REM TEPPICH(CHR$ 155)
9043 DATA "k",60,254,195,0,0,0,0
,0: REM TEPPICH(CHR$ 154)
9044 DATA "o",127,65,127,8,8,120
,56,120: REM SCHLUESSEL(CHR$ 158
)
9045 DATA "f",60,126,219,255,126
,60,36,195: REM FLOH(CHR$ 149)
9046 DATA "j",1,32,0,4,132,8,9,8
: REM BOMBE/LUNTE(CHR$ 153)
9050 RETURN
9098
9099 REM VARIABLEN

9100 LET k=2: LET l=17: REM
     LAUFVARIABLEN DES
     KUGELMONSTERS

9101 LET p=1: REM
     BILDZAEHLER

9102 LET z=0: REM
     PUNKTE-ZAEHLER PRO BILD

9103 LET z1=0: REM
     PUNKTE-ZAEHLER GESAMTES
     SPIEL

9105 LET b=7100: REM
     SPRUNGADRESSE BILDAUFBAU

9106 LET v=0: REM
     SPRUNGZAEHLER

9107 LET y=2: REM
     LAUFVARIABLE 2.TOTENKOPF

9108 LET g=60: REM
     SPRUNGADRESSE IN DAS
     JEWEILIGE BILDENTSPRECHENDE
     UNTERPROGRAMM. FESTLEGUNG
     IN 7000-7500.
     AUSSERDEM STEUERUNG DER
     TOTENKOEPFE IM 1.BILD

9109 LET x=6: REM
     LAUFVARIABLE 1.TOTENKOPF
```



```
9110 LET mv=0: REM
     STELLT FEST WIEVIEL SPIEL-
     FIGUREN VORHANDEN SIND

9112 LET m$="█████": REM
     FESTLEGUNG AB ZEILE 8000

9113 LET e=20: LET w=3: REM
     LAUFVARIABLEN DES FLOH's

9114 LET o=1: REM
     LAUFVARIABLE VOGEL 3.BILD
     UND TEPPICH 4.BILD

9115 LET t=0: REM
     FIGUREN-BEWEGUNG VOEGEL
     UND TEPPICHE

9116 LET c=0: REM
     LAUFVARIABLE VOGEL 4.BILD

9117 LET d=8: REM
     LAUFVARIABLE TEPPICH 5.BILD

9119 LET a=0: REM
     FIGUREN-BEWEGUNG FLOH

9120 LET n=0: REM
     FESTSTELLUNG OB ALLE
     BOMBEN IM WASSER SIND

9122 REM LET j=ZAEHLER IN ZEILE
     7520

9123 REM LET f=HEBEL GETROFFEN?

9124 REM LET q=AUF TREPPE
     SCHRITT WEITER GEHEN

9125 LET i=1: REM
     SPIELENDE WENN i=0

9126 LET ii=1: REM
     BOMBE EXPLODIERT WENN ii=0

9127 LET ru=1: REM
     IN DAS UNTERE BILD
     GEFALLEN WENN ru=0

9128 REM UDG's:
  CHR$ 144/145/146/147=ROLLTREPPE
  CHR$ 148/149=FLOH
  CHR$ 151     =HEBEL
  CHR$ 152/160=KUGELMONSTER
  CHR$ 153/152=BOMBE
  CHR$ 154/155=FLIEGENDER TEPPICH
  CHR$ 156/157=DACHZIEGEL
  CHR$ 158     =SCHLUESSEL
  CHR$ 159     =TEUFELKOPF
  CHR$ 161/162=VOGEL
  CHR$ 163     =SPRUNGFEDER
  CHR$ 164     =KLETTERSTANGE(u$)
  UND UMLAUTE
  CHR$ 144/158/162/164

9130 RETURN
9131
9200 REM UDG's BUCHSTABEN UND
     DACHZIEGEL
9201 RESTORE 9206
9202 FOR i=1 TO 7: READ q$
9203 FOR j=0 TO 7: READ c: POKE
USR q$+j,c: NEXT j: NEXT i
9206 DATA "u",40,0,68,68,68,68,5
6,0: REM UMLAUT UE(CHR$ 164)
9207 DATA "a",40,0,60,4,60,68,60
,0: REM UMLAUT AE(CHR$ 144)
9208 DATA "s",56,68,68,120,68,68
,120,64: REM UMLAUT SCHARFES "S"
(CHR$ 162)
9209 DATA "o",40,0,56,68,68,68,5
6,0: REM SCHLUESSEL(CHR$ 158)
9210 DATA "m",255,129,129,129,12
9,129,66,60: REM DACHPFANNE(CHR$
156)
```

```
9211 DATA "n",255,24,24,24,24,24
,36,195: REM DACHPFANNE(CHR$ 157
)
9212 DATA "e",60,126,219,255,126
,60,36,102: REM FLOH(CHR$ 148)
9214 RETURN
9499
9500 REM TITELBILD
9510 BORDER 5: PRINT AT 0,0;"1":
 PRINT #0;"1": CLS
9520 PAPER 3: INK 7: FOR i=0 TO
2: PRINT AT i,0;"
                          ": NEXT i: PRIN
T AT 1,0;" H G G   P R A E S E N
T I E R T "
9530 PAPER 2: PRINT AT 3,18;CHR$
 156: FOR i=19 TO 25: PRINT AT 3
,i;CHR$ 157: NEXT i: PRINT AT 3,
26;CHR$ 156: PRINT AT 4,17;CHR$
156: FOR i=18 TO 26: PRINT AT 4,
i;CHR$ 157: NEXT i: PRINT AT 4,2
7;CHR$ 156
9535 INK 0: PLOT 144,135: DRAW 0
,-38: PLOT 215,135: DRAW 0,-38
9540 PLOT 112,103: DRAW 0,-32: D
RAW 9,-18,1: DRAW -9,-45,-1: PLO
T 247,103: DRAW 0,-32: DRAW -9,-
18,-1: DRAW 9,-45,1
9545 PRINT AT 13,19; INK 5;"████
███"
9550 INK 5: FOR i=144 TO 151: PL
OT i,48: DRAW 15,15: NEXT i: FOR
 i=192 TO 199: PLOT i,63: DRAW 1
5,-15: NEXT i
9552 FOR i=16 TO 20: PRINT AT i,
18;"█";AT i,26;"█": NEXT i
9555 INK 0: PLOT 112,103: DRAW 1
1,0: DRAW 0,-7: DRAW 7,0: DRAW 0
,7: DRAW 11,0: DRAW 0,-7: DRAW 7
,0: DRAW 0,7: DRAW 11,0: DRAW 0,
-31
9557 PLOT 247,103: DRAW -10,0: D
RAW 0,-7: DRAW -7,0: DRAW 0,7: D
RAW -11,0: DRAW 0,-7: DRAW -7,0:
 DRAW 0,7: DRAW -11,0: DRAW 0,-3
1
9559 PLOT 171,96: DRAW 0,16: DRA
W 15,0,-3: DRAW 0,-16
9560 OVER 1: INK 3: PAPER 7: FOR
 i=160 TO 199 STEP 3: PLOT i,111
: DRAW 0,-16: NEXT i
9562 PLOT 160,106: DRAW 40,0: PR
INT AT 10,20;"█████"
9565 OVER 0: INK 0: PLOT 129,71:
 DRAW 0,9: DRAW 14,0,-3: DRAW 0,
-9: DRAW -14,0: PLOT 217,71: DRA
W 0,9: DRAW 14,0,-3: DRAW 0,-9:
DRAW -14,0
9570 PRINT AT 21,13; INK 4; PAPE
R 7;"▄";AT 21,14; INK 4; PAPER 2
;"████████████████";AT 21,31; I
NK 4; PAPER 7;"▄"
9575 INK 1: PLOT 160,8: DRAW 0,3
2: DRAW 39,0,-3: DRAW 0,-32: PLO
T 162,8: DRAW 0,30: DRAW 35,0,-3
: DRAW 0,-30
9580 PLOT 171,52: DRAW 0,-38: DR
AW -7,-5: PLOT 184,52: DRAW 0,-4
0: DRAW 10,-3
9590 PRINT AT 6,1;"    JUMP    "
9591 PRINT AT 7,1;"            "
9592 PRINT AT 8,1;"            "
9593 PRINT AT 10,2; INK 3; PAPER
 6;" A B O U T ": GO SUB 9594: G
O TO 9599
9594 LET mm=20: LET nn=17: FOR j
=1 TO 20: FOR i=1 TO 2: LET mm=m
m-1: LET nn=nn+1: GO SUB 9595: P
RINT AT mm+1,3;" ": PRINT AT mm,
3;CHR$ 148: PRINT AT nn-1,5;" ";
AT nn,5;CHR$ 148: BEEP .01,i: NE
XT i: NEXT j: PRINT AT 17,5;" ";
AT 19,3;" ";AT 16,2;"Spiel-";AT
18,2;"Erkl";CHR$ 144;"rung?";AT
20,3;CHR$ 148;" ";CHR$ 148;"  j/
```




```
n": GO TO 9599
9595 IF mm<18 THEN PRINT AT 18,3
;" ": LET mm=20: RETURN
9596 IF nn>20 THEN PRINT AT 21,5
;" ";AT 20,5;" ": LET nn=18: RET
URN
9597 RETURN
9600 IF INKEY$="n" THEN PRINT #0
; INK 7; PAPER 1;"ok, Moment bit
te!": PAPER 7: GO TO 9
9601 IF INKEY$="j" THEN GO TO 96
05
9602 GO TO 9600
9603
9604 REM SPIEL-ERKLAERUNG
9605 INK 0: PAPER 6: CLS
9607 PRINT AT 1,1;"Ein kleiner F
loh und sein          Freund, leben
slustig und froh,  haben ein sch
weres Schicksal zu meistern.
                       Sie sind beim
 herumh";CHR$ 164;"pfen un-    be
wu";CHR$ 162;"t in den Einflu";C
HR$ 162;"bereich      eines Magier
s geraten."
9610 PRINT AT 8,1;"Zur Strafe so
llen sie im Keller Gew";CHR$ 158
;"lbe der Behausung des         Ma
giers eingesperrt werden.       Do
ch siehe da: Nur bei einem      Fl
oh klappt der Zauber.           Im
merhin: Auch der Zweite ist     im
 Bann des Magiers, denn er      be
findet sich im Erdgescho";CHR$ 1
62;"       des Zauber-Reiches, ohn
e einen   sichtbaren Ausgang.
          Nun gilt es f";CHR$ 164
;"r diesen kleinen  Floh f";CHR$
 164;"r sich und seinen Freund e
inen Ausweg zu finden!"
9615 PRINT #0;" bitte Taste drüc
ken!"
9617 PAUSE 0: CLS : PRINT AT 1,1
;"Das Spiel f";CHR$ 164;"hrt dur
ch 6 Bilder      und ist beendet, we
nn der             zweite Floh aus dem
 Keller            befreit worden ist.

                   Abschließend ersche
int eine           HIGH-SCORE-Tabelle.

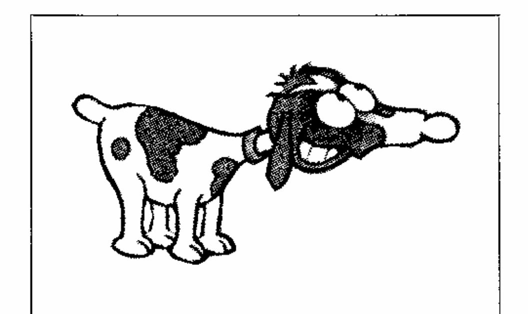
                   W";CHR$ 144;"hrend
des Spieles wird der        Punktest
and jeweils beim            Bildwech
sel als aktueller           Zwischen
stand angezeigt."
9619 PRINT AT 14,1;"Bei der Name
n-Eingabe können      mittels der
Grafik-Taste auch     Umlaute gesc
hrieben werden:
                            UDG ""a"" =
 ";CHR$ 144;"      UDG ""u"" = ";C
HR$ 164;"          UDG ""o"" = ";CHR
$ 158;"      UDG ""s"" = ";CHR$ 16
2;"": PRINT #0;"    bitte Taste d
rücken!"
9620 PAUSE 0: CLS : PRINT AT 3,3
;"TASTENBEDIENUNG:";AT 8,3;"n =
links";AT 10,7;"zum Springen zus
";CHR$ 144;"tzlich";AT 11,7;"CAP
S SHIFT";AT 13,3;"m = rechts";AT
 15,7;"zum Springen zus";CHR$ 14
4;"tzlich";AT 16,7;"CAPS SHIFT";
AT 18,3;"z = hochspringen"
9621 PRINT #0;"    bitte Taste dr
ücken!"
9622 PAUSE 0: PRINT #1;"      "; IN
K 7; PAPER 2;"Moment bitte!": PA
PER 7: GO TO 9
9899
9900 REM SPIEL-ENDE
9903 IF NOT ii THEN CLS : GO TO
9936
9905 LET z1=z1+(120-z)+400
9910 INK 1: FOR i=20 TO 11 STEP
-1: PRINT AT i,0;"██████████████
████████████████████": BEEP .01,17
: NEXT i
9912 INK 7: PAPER 1: PRINT AT 12
,1;"DANK DEINER HILFE HAT ES UNS
ER";AT 14,1;"FLOH GESCHAFFT SEIN
EN FREUND";AT 16,1;"ZU BEFREIEN.
";AT 18,1;"DIESER GANG FUEHRT IN
 DIE";AT 20,1;"FREIHEIT."
9914 GO SUB 9200
9915
9916 REM HIGHSCORE-TABELLE
9920 INPUT "BITTE NAMEN EINGEBEN
! ";n$
9922 CLS : BORDER 5: PAPER 5: IN
K 0: FOR i=0 TO 21: PRINT AT i,0
;"
  ": NEXT i: GO SUB 7180: PRINT
AT 2,5; PAPER 7; INK 2;" H I G H
 - S C O R E "
9924 IF z1>m6 THEN LET m8=m7: LE
T m7=m6: LET m6=z1: LET e$=d$: L
ET d$=c$: LET c$=n$: GO TO 9930
9926 IF z1>m7 THEN LET m8=m7: LE
T m7=z1: LET e$=d$: LET d$=n$: G
O TO 9930
9928 IF z1>m8 THEN LET m8=z1: LE
T e$=n$
9930 PRINT AT 6,5;"1. ";c$;AT 6,
21;"   ";m6
9932 PRINT AT 8,5;"2. ";d$;AT 8,
21;"   ";m7
9934 PRINT AT 10,5;"3. ";e$;AT 1
0,21;"   ";m8
9936 PRINT AT 13,14;"* * *";AT 1
7,7;"NEUES SPIEL?  j/n"
9937 PAPER 7: INK 0
9938 IF INKEY$="j" THEN CLS : GO
 TO 5
9939 IF INKEY$="n" THEN STOP
9940 GO TO 9938
```



FUCHS UND HUND

Für Colour Genie mit neuen Rom-Modulen

Man hat es schon nicht leicht als Fuchs, wenn man drei hungrige Junge im Fuchsbau hat, die gefüttert werden wollen.
Man weiß nie, woher das nächste Mahl kommen soll.
Aber diesmal wird es ganz bestimmt eines der Hühner aus dem Stall von Bauer Emsig sein – dort auf der anderen Seite des Flusses.
Das Problem ist nur, daß Fiffi, der treue Bullterrier des Bauern, in der



Nähe des Stalles schläft, und sobald die Hühner nur einen Laut von sich geben, wacht er auf und geht auf Spurensuche.
Doch du, als schlauer Fuchs, kannst Fiffi hereinlegen, indem du falsche Fährten auslegst, und während Fiffi sich darin verliert, ein Huhn erbeuten.
Das Spiel hat 10 Spielstufen, abhängig von der Anzahl deiner Fuchsbauten, in die du dein Beutegut ablieferst.
Weitere Angaben stehen in der Spielanleitung im Listing.
Das Programm belegt 4,5 K Speicherplatz.

```
COLOUR GENIE

COLOUR GENIE
5 '     - Fuchs und Hund -

  Geschrieben von W. und G. Brockel
10 CHAR2:GOSUB1120
20 CLEAR250
30 HU=130:FU=129:BA=131:CH=128
40 DATA40,-40,1,-1
45 '
   ## BEDIENUNGSANLEITUNG ##

50 CLS:PRINTTAB(7)"- Fuchs und Hund -
60 PRINTSTRING$(40,"-");
70 PRINT"Du must als Fuchs versuchen, so viele":POKE17493,8
80 PRINT"Huhner wie moglich aus dem Huhnerstall":POKE17529,7:POKE17540,6:POKE17556,7
90 PRINT"des Bauern zu stehlen. Er wird von einem";
100 PRINT"Wachhund beschutzt, der einen guten":POKE17622,7
110 PRINT"Geruchssinn hat.
120 PRINT"Bevor du ein neues Huhn bekommen kannst,";
130 PRINT"must du zuerst dein erbeutetes Huhn zu":POKE17730,8
140 PRINT"deiner Hohle bringen, aber der Hund ist":POKE17776,6
150 PRINT"auf deiner Spur !":PRINT
160 PRINT"Fuchs:"CHR$(FU),"Hund:"CHR$(HU),"Deine Spur:.
170 PRINT"Hohle:^","Huhn:"CHR$(CH),"Baum:"CHR$(BA):POKE17929,6
180 PRINT§560,;:INPUT"Schwierigkeitsstufe (1-10) - 1=schwer ";SL
190 IFSL<1ORSL>10GOTO180
200 PRINT:PRINT"Weitere Wahlmoglichkeit :":POKE18101,6:PRINT"-1- Hund immer unsichtbar
210 PRINT"-2- Hund nur sichtbar auf deiner Spur
220 PRINT"-3- Hund nur unsichtbar auf deiner Spur
230 PRINT"-4- Hund immer sichtbar
240 INPUTDL
250 IFDL<1ORDL>4GOTO240
260 ONDLGOSUB270,280,290,300:GOTO310
270 BB=246:FF=246:RETURN
280 BB=246:FF=HU:RETURN
290 BB=HU:FF=246:RETURN
300 BB=HU:FF=HU
310 HS=5:N$="Computer"
315 '
    ## VARIABELN ##

320 L$=STRING$(3,132):LI$=STRING$(3,132):BL$=STRING$(3,246)
330 CC$(1)=CHR$(132)+L$+BL$+L$+BL$+L$+CHR$(132)
340 CC$(2)=CHR$(132)+BL$+BL$+BL$+BL$+BL$+CHR$(132)
350 CC$(3)=CHR$(132)+LI$+BL$+LI$+BL$+LI$+CHR$(132)
360 R1$=STRING$(7,133):R2$=" ":RI$=R1$+R2$+R1$+R2$+R1$+R2$+R1$+R2$+R1$
370 SC=0:SF=1:DM=-1:CF=0:DG=17833
375 '
   ## SPIELFELD ZEICHNEN ##

380 CLS
390 FORA=1TO50-SL:POKERND(289)+17408,131:NEXT
400 FORA=17447TO18047STEP40:POKEA,148:NEXT
410 FORA=1TO3:PRINT§(A*40)+441,CC$(A)
420 NEXT
430 POKE17930,CH:POKEDG,HU
440 PRINT§280,RI$
450 FORA=1TOSL
460 BU=RND(199)+17408:IF(PEEK(BU-1))AND(PEEK(BU+1))<>131THEN460
470 POKEBU,94
480 NEXT
490 PRINT§320,"Flus":POKE17731,8:PRINT§682,"Punkte: 0":PRINT§640,STRING$(40,"-")
500 PRINT§762,"Punkterekord"HS"gehalten von "N$
510 PRINT§883,STRING$(35,244):PRINT§922,CHR$(238)"     F U C H S  U N D  H U N D
"CHR$(245);:PRINT§963,STRING$(35,229);
520 FORA=1TO1500:NEXT
530 PRINT§320,BL$+BL$
540 U=RND(639)+17408:IFPEEK(U)<>246THEN540ELSEPOKEU,FU
545 '
    ## SPIELBEGINN ##

550 PRINT§363,STRING$(36,246);
555 '
    ## FLUSS ##

560 W=W+1:W=WAND1:IFW=1THENPOKE&HF42B,68:POKE&HF42C,17ELSEPOKE&HF42B,17:POKE&HF42C,68
565 '
    ## BEWEGUNG DES FUCHSES ##

570 IFPEEK(-1984)=8MU=-40:GOTO620
580 IFPEEK(-1984)=16MU=40:GOTO620
590 IFPEEK(-1984)=32MU=-1:GOTO620
600 IFPEEK(-1984)=64MU=1:GOTO620
610 GOTO700
620 IFU+MU<17408ORU+MU>18047THEN700
630 PP=PEEK(U+MU)
640 IFPP=32ORPP=246ORPP=46GOTO690
650 IFU+MU=DGGOTO1030
660 IFPP=CHGOSUB890:GOTO700
670 IFPP=94ANDCF=1GOSUB990:GOTO700
680 GOTO700
690 POKEU,46:U=U+MU:POKEU,FU
695 '
    ## BEWEGUNG DES HUNDES AUF DER SUCHE
700 IFFL=1THEN790
710 IFDG+DM<17408ORDG+DM>18047GOSUB940:GOTO560
720 IFRND(10)=7GOSUB940
730 PP=PEEK(DG+DM)
740 IFPP=246GOTO780
750 IFPP=46THEN790
760 IFPP=FUTHEN1030
770 GOSUB940:GOTO560
780 IFSF=1THEN560ELSEPOKEDG,246:DG=DG+DM:POKEDG,BB:GOTO560
785 '
    ## HUND IST AUF DER SPUR ##
```



```
790 FL=1
800 PLAY(1,7,RND(5),11):FORI=1TO10:NEXT:PLAY(1,1,1,0)
810 RESTORE
820 FORX=1TO4:READDM
830 PP=PEEK(DG+DM)
840 IFPP=46THEN880
850 IFPP=FUTHEN1030
860 IFPP()46NEXT
870 FL=0:GOTO560
880 POKEDG,246:POKEDG+DM,FF:DG=DG+DM:GOTO560
885 '
    ** FUCHS ERBEUTET HUHN **


890 SF=0
900 IFPEEK(17930)=CHTHENPOKE17930,246ELSEPOKE17944,246
910 CF=1
920 SOUND1,0:SOUND12,100:SOUND13,9:SOUND8,31:FORJ=255TO120STEP-
1:SOUND0,J:NEXT
930 RETURN
935 '
    ** NEUE RICHTUNG FUER DEN HUND **


940 V=RND(4):ONVGOTO950,960,970,980
950 DM=-40:RETURN
960 DM=40:RETURN
970 DM=-1:RETURN
980 DM=1:RETURN
985 '
    ** HUHN IM FUCHSBAU ABLIEFERN **
990 SC=SC+1:CF=0:IFRND(2)=1THENPOKE17930,CHELSEPOKE17944,CH
1000 FORI=1TO6:PLAY(1,I,2,15):NEXT:PLAY(1,1,1,0)
1010 PRINT§682,"Punkte:"SC
1020 RETURN
1025 '
    ** FUCHS VOM HUND GEFANGEN **

1030 SOUND1,13:SOUND8,31:SOUND12,255:SOUND13,9:POKEDG,246:POKEU,HU:FORA=1TO255:SOUND0,A:NEXT:PRINT§880,CHR$(31)
1040 PRINT§845,"Du hast"SC:PRINT§856,"Huhner gefangen":POKE18265,7
1050 IFSC>HSLETHS=SC:PRINT§885,"und hast einen Punkterekord!":PRINT§920,"Gib deinen Namen ein:";:INPUTN$
1060 PRINT§774,HS"gehalten von "N$CHR$(30);
1070 PRINT§880,CHR$(31);"
Leertaste fur ein neues Spiel (E=Ende)";:POKE18339,7
1080 A$=INKEY$
1090 IFA$=" "GOTO370
1100 IFA$()"E"THEN1080ELSEPRINT"

Danke furs Spiel, "N$:POKE18335,7
1110 END
1115 '
    ** DATA FUER ZEICHENDEFINITION **

1120 DATA32,208,112,63,62,28,8,8
1130 DATA9,9,31,51,127,255,63,15
1140 DATA18,50,62,73,255,231,66,60
1150 DATA56,124,254,254,124,16,16,56
1160 DATA34,34,255,34,34,34,255,34
1170 DATA195,156,0,68,17,0,156,195
1180 RESTORE:FORJ=1TO4:READX:NEXT
1190 FORJ=&HF400TO&HF42F:READX:POKEJ,X:NEXT
1200 RETURN
```

Backe Backe Kuchen ... Setzen Sie Ihre Bäkkermütze auf, denn hier kommt ... die verrückte Kuchenbäckerei

Das Spiel ist in LASERBASIC für den LASER 2001 geschrieben und wird mit dem linken Joystick gesteuert. Zur Steuerung kann auch ein ATARI Joystick (oder ähnlicher) benutzt werden, da im Laufe des Spieles kein Feuerknopf angesprochen wird.

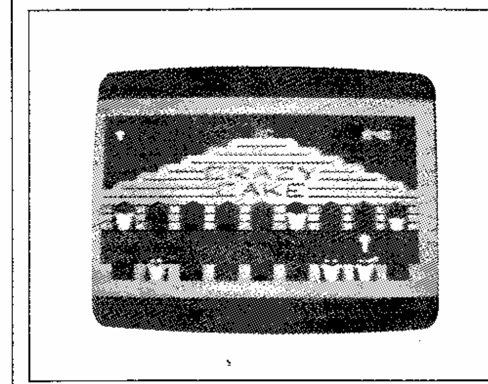
Doch nun zum Spiel:
Sie sind Bäcker in einer Kuchenbäckerei und haben die Aufgabe, für ein großes Fest Kuchen zu backen. In Ihrem großen Backofen sind immer 9 Kuchen gleichzeitig. Zu Beginn einer jeden Runde liegt natürlich nur der Teig der Kuchen in den einzelnen Türen des Backofens. Dieser Teig fängt nun im Backofen an aufzugehen, und das natürlich ungleichmäßig. Sie müssen jetzt die Kuchen mit Ihrer Bäckerschaufel aus dem Backofen herausholen, sobald diese fertig gebacken sind. Einen fertig gebackenen Kuchen erkennen Sie an der Farbe, denn er wechselt diese von gelb nach leicht rot. Lassen Sie die Kuchen zu lange im Backofen, so verbrennen diese. Holen Sie die Kuchen jedoch zu früh aus dem Backofen, so fallen diese zusammen, und der Rest des zusammengefallenen Kuchens verbrennt ebenfalls im Backofen. Wenn Sie es geschafft haben einen Kuchen heil aus dem Backofen herauszuholen, so erscheint dieser in dem zugehörigen Kästchen unter dem Backofen. Doch die fertig gebackenen Kuchen, die Sie bereits aus dem Backofen herausgeholt haben, sind noch lange nicht in Sicherheit, denn in die Backstube hat sich eine sehr gefräßige Maus eingeschlichen, die Ihnen Ihre fertig gebackenen Kuchen wegfressen will. Sie könnten jetzt natürlich die Maus erschlagen, oder eine Mausefalle aufstellen. Da Sie natürlich ein großer Tierfreund sind, werden Sie nichts dergleichen tun, sondern die Maus mit Käse füttern, damit diese Ihren Kuchen nicht frißt, denn solange die Maus Käse zu fressen hat, wird sie Ihren Kuchen in Ruhe lassen.
Eine Runde des Spieles ist immer dann beendet, wenn keine Kuchen im Backofen vorhanden sind, die noch weiter backen können. Die Punkte werden auch immer erst zum Rundenende abgerechnet. Wenn es Ihnen gelingt mindestens 5 Kuchen zu „retten", so geht das Spiel mit einem höheren Schwierigkeitsgrad weiter. Gelingt es Ihnen jedoch nicht, mindestens 5 Kuchen heil aus dem Backofen zu bringen, so wird Ihnen eine Ihrer anfänglich 3 Bäckerschaufeln abgenommen, und das Spiel geht mit dem gleichen Schwierigkeitsgrad und 9 neuen Kuchen weiter, sofern Sie noch eine Bäkkerschaufel übrig haben. Das Spiel ist dann beendet, wenn Ihnen alle Bäkkerschaufeln abgenommen wurden. Sollte es Ihnen gelingen in einer Runde sämtliche 9 Kuchen heil aus dem Backofen zu bringen, und vor der Maus zu „retten", so bekommen Sie zur Belohnung eine Bäckerschaufel geschenkt.

Sie haben die Auswahl zwischen 4 Spielversionen:
1 – Bäckerlehrling
(Beginn bei Schwierigkeit 1)
2 – Bäckergeselle
(Beginn bei Schwierigkeit 3)
3 – Bäckermeister
(Beginn bei Schwierigkeit 5)
4 – Chef der Bäckerei
(Beginn bei Schwierigkeit 7)
Die Schwierigkeit wird bis auf maximal Stufe 8 erhöht. Danach bleibt der Schwierigkeitsgrad konstant.

Steuerung:
Das Spiel wird mit dem linken Joystick gesteuert.
Eine Joystickbewegung nach links oder rechts bewegt die Bäckerschaufel in die entsprechende Richtung, und zwar immer um eine Ofentür. Um einen Kuchen aus dem Ofen zu holen, wird der Joystick nach oben gedrückt. Wird der Joystick nach unten gezogen, so werden immer zwei Käsestückchen in die Laufbahn der Maus gelegt, und zwar immer unter der Bäckerschaufel.

Die Anzeige:
Links oben im Bild sind die Anzahl der Reserve-Bäckerschaufeln angezeigt. Zu Spielbeginn zeigt diese Anzeige 2 Schaufeln an, das bedeutet, daß Sie die beiden Reserveschaufeln und die zur Zeit benutzte Schaufel zur Verfügung haben (also 3 Schaufeln).
Rechts oben im Bild ist die Punktanzeige eingeblendet (graue Zahlen). Diese Anzeige wird immer erst zum Ende einer Runde aktualisiert. Die Anzeige zeigt also während des gesamten Verlaufes einer Runde keine Veränderung an.
Die grüne Anzeige mitten über dem Backofen zeigt an, wieviel Punkte Sie in dieser Runde für jeden „geretteten" Kuchen bekommen.

Tips zum Spiel:
Die Maus läuft von Anfang an immer auf dem Bildschirm hin und her. Füttern Sie die Maus deshalb von Anfang an mit Käse, und nicht erst, wenn Sie einen Kuchen aus dem Ofen geholt haben, denn wenn die Maus gerade, während Sie einen Kuchen aus dem Backofen holen, unter der Bäckerschaufel steht, und kein Käse über dem entsprechenden Feld liegt, die Maus sofort den Kuchen frißt. Die Maus frißt immer erst den Käse, bevor sie sich am Kuchen vergreift. Wenn ein Kuchen in einem Feld steht, und über diesem Feld Käse liegt, so wird der Käse gefressen. Läuft die Maus jedoch ein zweitesmal über dieses Feld, so wird der Kuchen gefressen, es sei denn, es wurde wieder neuer Käse über das Feld gelegt. Seien Sie deshalb besonders bei dem rechten und linken Eckfeld auf der Hut, da die Maus hier die Richtung wechselt.



Tips bei der Eingabe des Programms:
Benutzen Sie bei der Eingabe nach Möglichkeit die CTRL-Tasten. Damit werden die Tippfehler bei der Eingabe um einen Großteil verringert.
Bevor Sie das Programm zum erstenmal starten, SAVEN Sie das Programm unbedingt mehrmals auf Cassette oder Diskette. Das Programm belegt den Speicher fast vollständig, was bei bestimmten Eingabefehlern beim LASER 2001 zum Systemabsturz führen kann. In manchen Fällen ist dadurch dann auch die RESET-Taste unwirksam.

Anmerkung:
Das Spiel sollte unbedingt mit einem Farbfernseher oder Farbmonitor gespielt werden. Bei Verwendung eines Schwarz-Weiß-Monitors hat das Spiel keine Wirkung, da die Farbtöne eine fast einheitliche graue Fläche ergeben, und der Farbwechsel der Kuchen nicht richtig wahrgenommen wird. Der Fernseher sollte möglichst genau auf den LASER eingestellt sein (am besten Kanal 4 am Schalter auf der Unterseite einstellen), da sonst an Stelle der schönen akkustischen Untermalung nur ein fürchterliches Gejaule zu hören ist.
Doch nun machen Sie sich auf den Weg in die verrückte Kuchenbäckerei. Ich wünsche ein ERRORfreies Eintippen und viele schöne Kuchen bei ... Backe Backe Kuchen ... CRAZY CAKE.

```
1  REM ************************
2  REM *  C R A Z Y   C A K E  *
3  REM *----------------------*
4  REM *      LASER 2001       *
5  REM *  und linken Joystick  *
6  REM *----------------------*
7  REM * (c)1984 by Tronicsoft *
8  REM *           V.Becker    *
9  REM ************************
10  REM
70  CLEAR
80  DIM KF(9,2),MF(28,2)
85  GOSUB 20000
90 ST = 1:SM = 223:KU = 0:PA = 28
92 PK = 0:XS = 0
100  GR
110  GOSUB 50000
111  COLOR= 2,4
114 BS = 2
116  GOSUB 25000
120 S = 127:XA = 220
125  GOSUB 31000
130  GOSUB 58000
140 P$ =  STR$ (20 * (RU + 1)):XA = 136: COLOR= 2,4: GOSUB 31007
890 P = 4
900  FOR I = 1 TO 9:KF(I,1) = 1:KF(I,2) = P * 8 - 1:P = P + 3: NEXT I
905  COLOR= 10,4:ZK = 126
910  FOR I = 1 TO 9:SK = KF(I,2): GOSUB 48270: NEXT I
920 RU = RU + 1: IF RU > 8 THEN RU = 8
930 VV = 5
940  FOR K = 1 TO 2: FOR I = 1 TO 28:MF(I,K) = 0: NEXT I: NEXT K
950  GOSUB 61000
1000 A =  INT ( RND (255) * 8 + 1.5): IF KF(A,1) = 9 THEN  GOTO 1000
1010 KF(A,1) = KF(A,1) + 1:ZK = 126:SK = KF(A,2)
1020  COLOR= 10,4
1030  ON KF(A,1) GOSUB 48270,48210,48150,48110,48070,48040,48000,47900,47950
1500  FOR LI = 1 TO 9 - RU
2000  CALL  - 1536
2005  ON ST GOSUB 49000,49100
2010 JS =  PEEK (24)
2020  IF JS = 0 THEN  GOTO 3000
2025  IF JS > 100 THEN  GOTO 3000
2030  GOTO 10000
3000  NEXT LI
3005  FOR I = 1 TO 9: IF KF(I,1) <  > 9 THEN  GOTO 3010
3006  NEXT I
3007  GOTO 4000
3010  GOTO 1000
4000 PK = PK + 20 * KU * RU
4005  GOSUB 59000
4010 XA = 220: GOSUB 31000
4012  IF KU < 5 THEN  GOSUB 24000:BS = BS - 1:RU = RU - 1
4014  IF KU = 9 THEN BS = BS + 1: GOSUB 25000
4016  IF BS < 0 THEN  GOTO 5000
4020  COLOR= 7,4
4023  COLOR= 8,4: GOSUB 50170
4025  GOSUB 49200
4030  FOR I = 3 TO 27 STEP 3
4040  FOR K = 0 TO 15
4050  UNPLOT I * 8 + K,156 TO I * 8 + K,175
4060  NEXT K
4065  SOUND (5 * I - 10,2,15),(5 * I - 15,2,15)
4070  NEXT I
4080 KU = 0
4090  GOTO 120
```

```
5000  COLOR= 2,4
5010  FOR I = 100 TO 127: PLOT 16,I TO 239,I
5020  NEXT I
5025 Z = 105
5030  FOR K = 1 TO 8
5040  READ S,SK
5050  FOR I = 0 TO 2
5060  ON SK GOSUB 51000,52000,53000,54000,55000,56000,57000,41000,42000,43000,44
000
5070  NEXT I
5080  NEXT K
5090  GOSUB 61000
5100  GET A$: RUN
10000 JS = JS / 2
10010  IF JS > 10 THEN  GOTO 11000
10020  ON JS GOTO 10500,10100,10100,10300,3000,10100
10100  IF S >  = 223 THEN  GOTO 3000
10110  GOSUB 59000
10120 S = S + 24:VV = VV + 1: GOSUB 58000
10130  GOTO 3000
10200  IF S <  = 31 THEN  GOTO 3000
10210  GOSUB 59000
10220 S = S - 24:VV = VV - 1: GOSUB 58000
10230  GOTO 3000
10300  IF KF(VV,1) = 9 THEN  GOTO 3000
10302 SK = S
10305  GOSUB 59000
10310  ON ST GOSUB 49000,49100
10320  GOSUB 59200
10330  ON ST GOSUB 49000,49100
10340  IF KF(VV,1) = 8 THEN  COLOR= 4,4: GOSUB 59400
10350  ON ST GOSUB 49000,49100
10360  IF KF(VV,1) = 8 THEN  COLOR= 9,4:ZK = 175: GOSUB 48000:ZK = 126:KF(VV,1)
= 9:KU = KU + 1:MF(((S + 1) / 8) - 2,2) = 1:MF(((S + 1) / 8) - 1,2) = 1
10370  ON ST GOSUB 49000,49100
10380  IF KF(VV,1) < 8 THEN  GOSUB 59300:KF(VV,1) = 9
10390  ON ST GOSUB 49000,49100
10400  GOSUB 59500
10410  ON ST GOSUB 49000,49100
10420  GOSUB 58000
10430  ON ST GOSUB 49000,49100
10440  GOTO 3000
10500  COLOR= 11,4
10510  FOR KS = 156 TO 157: PLOT S - 3,KS TO S - 2,KS
10520  PLOT S + 2,KS TO S + 3,KS
10530  NEXT KS
10535  SOUND (160 - (KS - 156) * 20,2,15),(150 - (KS - 156) * 20,2,15)
10536 WT =  INT ((S + 1) / 8) - 1:MF(WT,1) = 1:MF(WT - 1,1) = 1
10540  GOTO 3000
11000 JS = JS - 15
11010  ON JS GOTO 10200,10200,3000,3000,10200
20000  HOME
20005  TEXT
20010  COLOR= 1,2
20020  PRINT
20030  PRINT  SPC( 8);"C R A Z Y - C A K E"
20040  PRINT  SPC( 8);"-------------------"; CHR$ (24); CHR$ (24); CHR$ (24)
20050  PRINT  SPC( 8);"Spielversionen :"; CHR$ (24); CHR$ (24)
20060  PRINT  SPC( 8);"1 Baeckerlehrling"; CHR$ (24)
20070  PRINT  SPC( 8);"2 Baeckergeselle"; CHR$ (24)
20080  PRINT  SPC( 8);"3 Baeckermeister"; CHR$ (24)
```

```
20090  PRINT  SPC( 8);"4 Chef der Baeckerei"; CHR$ (24)
20110  FOR I = 1 TO 6: PRINT  CHR$ (24);: NEXT I
20120  PRINT  SPC( 8);"(c) 1984 by Tronicsoft";
20125  GOSUB 61000
20130  FOR I = 1 TO 4: PRINT  CHR$ (5);: NEXT I: PRINT
20140  PRINT  CHR$ (5); SPC( 8);"bitte waehlen Sie : ";
20150  INPUT RU
20160 RU =  INT (RU): IF RU < 1 OR RU > 4 THEN  GOTO 20140
20170 RU = (RU - 1) * 2
20180  PRINT  CHR$ (5);" bitte den linken Joystick bewegen"
20190  CALL  - 1536: IF  PEEK (24) = 0 THEN  GOTO 20190
20200  RETURN
24000  COLOR= 14,4
24110  UNPLOT 8 * BS + 16,33
24120  UNPLOT 8 * BS + 16 + 1,32 TO 8 * BS + 16 + 1,34
24130  UNPLOT 8 * BS + 16 + 2,31 TO 8 * BS + 16 + 2,39
24140  UNPLOT 8 * BS + 16 + 3,31 TO 8 * BS + 16 + 3,39
24150  UNPLOT 8 * BS + 16 + 4,32 TO 8 * BS + 16 + 4,34
24160  UNPLOT 8 * BS + 16 + 5,33
24180  RETURN
25000  COLOR= 14,4
25100  FOR I = 1 TO BS
25110  PLOT 8 * I + 16,33
25120  PLOT 8 * I + 16 + 1,32 TO 8 * I + 16 + 1,34
25130  PLOT 8 * I + 16 + 2,31 TO 8 * I + 16 + 2,39
25140  PLOT 8 * I + 16 + 3,31 TO 8 * I + 16 + 3,39
25150  PLOT 8 * I + 16 + 4,32 TO 8 * I + 16 + 4,34
25160  PLOT 8 * I + 16 + 5,33
25162  SOUND (135,2,15),(134,2,12),(136,2,12)
25164  SOUND (170,2,15),(169,2,12),(171,2,12)
25166  SOUND (195,2,15),(194,2,12),(196,2,12)
25168  SOUND (202,2,15),(201,2,12),(203,2,12)
25170  NEXT I
25180  RETURN
30000  FOR I = 31 TO 39 STEP 8: PLOT SP + 2,I TO SP + 4,I
30010  NEXT I
30020  FOR K = 1 TO 5 STEP 4
30030  FOR I = 32 TO 36 STEP 4: PLOT SP + K,I TO SP + K,I + 2
30040  NEXT I
30050  NEXT K
30060  RETURN
30100  PLOT SP + 5,32 TO SP + 5,34
30110  PLOT SP + 5,36 TO SP + 5,38
30120  RETURN
30200  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30210  NEXT I
30220  PLOT SP + 5,32 TO SP + 5,34
30230  PLOT SP + 1,36 TO SP + 1,38
30240  RETURN
30300  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30310  NEXT I
30320  PLOT SP + 5,32 TO SP + 5,34
30330  PLOT SP + 5,36 TO SP + 5,38
30340  RETURN
30400  PLOT SP + 1,32 TO SP + 1,34
30410  PLOT SP + 5,32 TO SP + 5,34
30420  PLOT SP + 2,35 TO SP + 4,35
30430  PLOT SP + 5,36 TO SP + 5,38
30440  RETURN
30500  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30510  NEXT I
30520  PLOT SP + 1,32 TO SP + 1,34
```

```
30530  PLOT SP + 5,36 TO SP + 5,38
30540  RETURN
30600  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30610  NEXT I
30620  PLOT SP + 1,32 TO SP + 1,34
30630  PLOT SP + 5,36 TO SP + 5,38
30640  PLOT SP + 1,36 TO SP + 1,38
30650  RETURN
30700  PLOT SP + 2,31 TO SP + 4,31
30710  PLOT SP + 5,32 TO SP + 5,34
30720  PLOT SP + 5,36 TO SP + 5,38
30730  RETURN
30800  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30810  NEXT I
30830  FOR K = 1 TO 5 STEP 4
30840  FOR I = 32 TO 36 STEP 4: PLOT SP + K,I TO SP + K,I + 2
30850  NEXT I
30860  NEXT K
30870  RETURN
30900  FOR I = 31 TO 39 STEP 4: PLOT SP + 2,I TO SP + 4,I
30910  NEXT I
30920  PLOT SP + 5,32 TO SP + 5,34
30930  PLOT SP + 5,36 TO SP + 5,38
30940  PLOT SP + 1,32 TO SP + 1,34
30950  RETURN
31000  COLOR= 14,4
31005 P$ =  STR$ (PK)
31007 SP = XA - 7 *  LEN (P$)
31010 L =  VAL ( MID$ (P$,1,1)) + 1
31020  UNPLOT SP + 1,31 TO SP + 5,31 TO SP + 5,39 TO SP + 1,39 TO SP + 1,31
31025  UNPLOT SP + 2,35 TO SP + 4,35
31030  ON L GOSUB 30000,30100,30200,30300,30400,30500,30600,30700,30800,30900
31035  IF  LEN (P$) <  = 1 THEN  RETURN
31040 P$ =  RIGHT$ (P$, LEN (P$) - 1)
31050  GOTO 31007
41000  UNPLOT S + 11 + I,Z + 4 - I TO S + I,Z + 4 - I TO S + I,Z + 15 - I TO S +
 11 + I,Z + 15 - I TO S + 11 + I,Z + 10 - I TO S + 3 + I,Z + 10 - I
41010  RETURN
42000  UNPLOT S + I,Z + 15 - I TO S + I,Z + 4 - I TO S + 5 + I,Z + 10 - I TO S +
 6 + I,Z + 10 - I TO S + 11 + I,Z + 4 - I TO S + 11 + I,Z + 15 - I
42010  RETURN
43000  UNPLOT S + I,Z + 4 - I TO S + 11 + I,Z + 4 - I TO S + 11 + I,Z + 15 - I T
O S + I,Z + 15 - I TO S + I,Z + 4 - I
43010  RETURN
44000  UNPLOT S + I,Z + 4 - I TO S + 5 + I,Z + 15 - I TO S + 6 + I,Z + 16 - I TO
 S + 11 + I,Z + 4 - I
44010  RETURN
47900  COLOR= 9,4
47910  GOSUB 48000
47920  RETURN
47950  GOSUB 59600
47960  GOSUB 59310
47970  RETURN
48000  PLOT SK - 5,ZK - 15 TO SK - 4,ZK - 15
48010  PLOT SK + 5,ZK - 15 TO SK + 6,ZK - 15
48020  PLOT SK - 6,ZK - 14 TO SK - 2,ZK - 14
48030  PLOT SK + 3,ZK - 14 TO SK + 7,ZK - 14
48040  PLOT SK - 6,ZK - 13 TO SK + 7,ZK - 13
48050  PLOT SK - 5,ZK - 12 TO SK + 6,ZK - 12
48060  PLOT SK - 4,ZK - 11 TO SK + 5,ZK - 11
48070  FOR KK =  - 10 TO  - 8: PLOT SK - 4,ZK + KK TO SK + 5,ZK + KK
```

```
48080  NEXT KK
48090  PLOT SK - 4,ZK - 7 TO SK - 4,ZK - 6
48100  PLOT SK + 5,ZK - 7 TO SK + 5,ZK - 6
48110  PLOT SK - 3,ZK - 7 TO SK + 4,ZK - 7
48120  PLOT SK - 3,ZK - 6 TO SK + 4,ZK - 6
48130  PLOT SK - 3,ZK - 5 TO SK - 3,ZK - 3
48140  PLOT SK + 4,ZK - 5 TO SK + 4,ZK - 3
48150  PLOT SK - 2,ZK - 5 TO SK + 3,ZK - 5
48160  PLOT SK - 2,ZK - 4 TO SK + 3,ZK - 4
48170  PLOT SK - 2,ZK - 3 TO SK - 2,ZK - 1
48180  PLOT SK + 3,ZK - 3 TO SK + 3,ZK - 1
48190  PLOT SK - 3,ZK - 2 TO SK - 3,ZK - 1
48200  PLOT SK + 4,ZK - 2 TO SK + 4,ZK - 1
48210  PLOT SK - 1,ZK - 3 TO SK + 2,ZK - 3
48220  PLOT SK - 1,ZK - 2 TO SK + 2,ZK - 2
48230  PLOT SK - 1,ZK - 1 TO SK - 1,ZK
48240  PLOT SK + 2,ZK - 1 TO SK + 2,ZK
48250  PLOT SK - 2,ZK
48260  PLOT SK + 3,ZK
48270  PLOT SK,ZK - 1 TO SK + 1,ZK - 1
48280  PLOT SK,ZK TO SK + 1,ZK
48290  RETURN
49000  COLOR= 7,4
49002  GOSUB 49200
49003 SM = SM - RU: IF SM <  = 31 THEN ST = 2
49004 PM =  INT (SM / 8) - 1: IF PA = PM THEN  GOTO 49010
49005  IF MF(PM,1) = 1 THEN MF(PM,1) = 0: GOTO 49010
49006  IF MF(PM,2) = 0 THEN  GOTO 49010
49007 MF(PM,2) = 0: COLOR= 4,4
49008  GOSUB 49500
49010  PLOT SM + 2,153 TO SM + 3,153 TO SM + 4,154 TO SM + 4,155 TO SM + 3,156 T
O SM - 2,156
49020  PLOT SM,155 TO SM + 2,155
49030  PLOT SM - 3,157 TO SM + 2,157
49040 PA = PM
49050  RETURN
49100  COLOR= 7,4
49102  GOSUB 49200
49103 SM = SM + RU: IF SM >  = 223 THEN ST = 1
49104 PM =  INT (SM / 8) - 1: IF PA = PM THEN  GOTO 49110
49105  IF MF(PM,1) = 1 THEN MF(PM,1) = 0: GOTO 49110
49106  IF MF(PM,2) = 0 THEN  GOTO 49110
49107 MF(PM,2) = 0: COLOR= 4,4
49108  GOSUB 49500
49110  PLOT SM - 1,153 TO SM - 2,153 TO SM - 3,154 TO SM - 3,155 TO SM - 2,156 T
O SM + 3,156
49120  PLOT SM - 1,155 TO SM + 1,155
49130  PLOT SM,157 TO SM + 4,157
49140 PA = PM
49150  RETURN
49200  FOR CL = 153 TO 157: UNPLOT SM - 3,CL TO SM + 4,CL
49210  NEXT CL
49230  RETURN
49500  IF MF(PM - 1,2) <  > 0 THEN P = PM - 1
49510  IF MF(PM + 1,2) <  > 0 THEN P = PM
49520 SP = P * 8 + 8
49530  FOR CL = 160 TO 175: UNPLOT SP,CL TO SP + 15,CL
49540  NEXT CL
49550  COLOR= 1,4:ZK = 175:SK = SP + 7: GOSUB 48150
49560  GOSUB 60000
49570 ZK = 126:KU = KU - 1
49575 MF(P,2) = 0:MF(P + 1,2) = 0
```

```
49577  COLOR= 15,4
49580  RETURN
50000  COLOR= 2,4
50005  FOR I = 0 TO 15
50010  PLOT I,I TO 255 - I,I TO 255 - I,191 - I TO I,191 - I TO I,I
50020  NEXT I
50025  COLOR= 13,4
50030  FOR I = 2 TO 29 STEP 3
50040  FOR K = 0 TO 7
50050  PLOT I * 8 + K,160 TO I * 8 + K,175
50060  NEXT K: NEXT I
50062  COLOR= 8,4
50065 SL = 14:SR = 17
50070  FOR I = 0 TO 6
50080  FOR K = 0 TO 7
50090 Z = (5 + I) * 8 + K
50100  PLOT SL * 8,Z TO SR * 8 + 7,Z
50120  NEXT K
50130 SL = SL - 2:SR = SR + 2
50132  IF I > 5 THEN  GOTO 50140
50135  FOR J = 1 TO 7: CIRCLE (SL * 8 + 14,Z),J: CIRCLE (SR * 8 - 7,Z),J: NEXT J
50140  NEXT I
50150  FOR I = 96 TO 127
50160  PLOT 16,I TO 239,I
50165  NEXT I
50167  GOSUB 55062
50170  FOR I = 3 TO 27 STEP 3
50180  FOR K = 0 TO 15
50190  UNPLOT I * 8 + K,104 TO I * 8 + K,126
50200  NEXT K
50210  UNPLOT I * 8 + 7,100 TO I * 8 + 8,100
50220  UNPLOT I * 8 + 5,101 TO I * 8 + 10,101
50230  UNPLOT I * 8 + 3,102 TO I * 8 + 12,102
50240  UNPLOT I * 8 + 1,103 TO I * 8 + 14,103
50250  NEXT I
50255  IF XS = 1 THEN  RETURN
50260  RESTORE
50270  FOR K = 1 TO 9
50280  READ S,Z,SK
50290  FOR I = 0 TO 1
50300  ON SK GOSUB 51000,52000,53000,54000,55000,56000,57000
50310  NEXT I
50320  NEXT K
50330 XS = 1
50340  RETURN
50900  DATA 89,64,1,105,64,2,121,64,3,137,64,4,153,64,5,97,80,1,113,80,3,129,80,
6,145,80,7
50950  DATA 56,8,72,3,88,9,104,7,136,10,152,11,168,7,184,2
51000  UNPLOT S + 11 + I,Z + 4 - I TO S + 0 + I,Z + 4 - I TO S + 0 + I,Z + 15 -
I TO S + 11 + I,Z + 15 - I
51010  RETURN
52000  UNPLOT S + 0 + I,Z + 15 - I TO S + 0 + I,Z + 4 - I TO S + 8 + I,Z + 4 - I
 TO S + 11 + I,Z + 7 - I TO S + 8 + I,Z + 10 - I TO S + 2 + I,Z + 10 - I TO S +
11 + I,Z + 15 - I
52010  RETURN
53000  UNPLOT S + 0 + I,Z + 15 - I TO S + 5 + I,Z + 4 - I TO S + 6 + I,Z + 4 - I
 TO S + 11 + I,Z + 15 - I
53010  UNPLOT S + 3 + I,Z + 10 - I TO S + 8 + I,Z + 10 - I
53020  RETURN
54000  UNPLOT S + 0 + I,Z + 4 - I TO S + 11 + I,Z + 4 - I TO S + 0 + I,Z + 15 -
I TO S + 11 + I,Z + 15 - I
54010  RETURN
```

```
55000  UNPLOT S + 0 + I,Z + 4 - I TO S + 5 + I,Z + 10 - I TO S + 5 + I,Z + 15 -
I TO S + 6 + I,Z + 15 - I TO S + 6 + I,Z + 10 - I TO S + 11 + I,Z + 4 - I
55010  RETURN
55062  COLOR= 1,4
55065 SL = 14:SR = 17
55070  FOR I = 0 TO 6
55090 Z = (5 + I) * 8 + 7
55100  PLOT SL * 8,Z TO SR * 8 + 7,Z
55130 SL = SL - 2:SR = SR + 2
55140  NEXT I
55150  FOR I = 95 TO 127 STEP 8
55160  PLOT 16,I TO 239,I
55170  NEXT I
55180  COLOR= 8,4
55190  PLOT 88,71 TO 165,71
55200  PLOT 88,79 TO 165,79
55210  PLOT 96,95 TO 157,95
55220  PLOT 96,87 TO 157,87
55230  RETURN
56000  UNPLOT S + 0 + I,Z + 4 - I TO S + 0 + I,Z + 15 - I
56010  UNPLOT S + 1 + I,Z + 10 - I TO S + 4 + I,Z + 10 - I
56020  UNPLOT S + 11 + I,Z + 4 - I TO S + 4 + I,Z + 10 - I TO S + 11 + I,Z + 15
- I
56030  RETURN
57000  UNPLOT S + 11 + I,Z + 4 - I TO S + 0 + I,Z + 4 - I TO S + 0 + I,Z + 15 -
I TO S + 11 + I,Z + 15 - I
57010  UNPLOT S + 1 + I,Z + 10 - I TO S + 6 + I,Z + 10 - I
57020  RETURN
58000  COLOR= 15,4
58010  PLOT S,132 TO S,147
58020  PLOT S + 1,132 TO S + 1,147
58030  PLOT S - 1,133 TO S - 1,138
58040  PLOT S + 2,133 TO S + 2,138
58050  PLOT S - 2,134 TO S - 2,137
58060  PLOT S + 3,134 TO S + 3,137
58070  RETURN
59000  COLOR= 15,4
59010  UNPLOT S + 3,134 TO S + 3,137
59020  UNPLOT S - 2,134 TO S - 2,137
59030  UNPLOT S + 2,133 TO S + 2,138
59040  UNPLOT S - 1,133 TO S - 1,138
59050  UNPLOT S + 1,132 TO S + 1,147
59060  UNPLOT S,132 TO S,147
59070  RETURN
59200  COLOR= 15,4
59210  PLOT S,129 TO S,143
59220  PLOT S + 1,129 TO S + 1,143
59230  PLOT S - 2,128 TO S + 3,128
59240  RETURN
59300  GOSUB 59400
59310  COLOR= 1,4: GOSUB 48150
59315  GOSUB 60000
59320  RETURN
59400  COLOR= 4,4
59410  FOR CL = 111 TO 126: UNPLOT S - 7,CL TO S + 8,CL
59420  NEXT CL
59430  RETURN
59500  COLOR= 15,4
59510  UNPLOT S,129 TO S,143
59520  UNPLOT S + 1,129 TO S + 1,143
59530  UNPLOT S - 2,128 TO S + 3,128
```

```
59540  RETURN
59600  COLOR= 4,4
59610  FOR CL = 111 TO 126: UNPLOT SK - 7,CL TO SK + 8,CL
59620  NEXT CL
59630  RETURN
60000  SOUND (255,1,0),(255,1,0),(255,1,0),(1,1,0,1)
60001  SGEN 159,191,223,255
60002  SOUND (120,1,0),(150,1,0)
60005  FOR I = 1 TO 10
60010  SGEN 50 + 20 * I,I,20 + 10 * I
60015  FOR K = 1 TO 5 * I: NEXT K
60020  NEXT I
60025  SOUND (255,1,0),(255,1,0),(255,1,0),(1,1,0,1)
60030  RETURN
61000  SOUND (128,10,15),(148,10,15)
61010  SOUND (128,10,15),(148,10,15)
61020  SOUND (135,10,15),(160,10,15)
61030  SOUND (135,10,15),(160,10,15)
61040  SOUND (128,20,15),(148,20,15)
61050  SOUND (95,10,15),(128,10,15)
61060  SOUND (1,10,15),(95,10,15)
61070  SOUND (128,10,15),(148,10,15)
61080  SOUND (128,10,15),(148,10,15)
61090  SOUND (135,10,15),(160,10,15)
61100  SOUND (135,10,15),(160,10,15)
61110  SOUND (128,20,15),(148,20,15)
61120  SOUND (95,20,15),(128,20,15)
61130  RETURN
```

Codewort für die Programme!

Sicherlich haben Sie schon länger den Wunsch, das eine oder andere Programm zu verschließen, so daß kein anderer damit arbeiten oder spielen kann, ausgenommen Sie selbst natürlich!
Man müßte einen SCHLÜSSEL haben, um ein Programm starten zu können ...
... ein Codewort muß es sein!
Ein Programm wird von Cassette oder Diskette geladen. Man will es schließlich mit RUN starten, was passiert jetzt?
Der Computer gibt auf dem Bildschirm folgenden Text aus:

BITTE CODEWORT EINGEBEN

Sie schreiben das Codewort; auf dem Bildschirm sieht man das Codewort nicht, sondern eine Reihe von diesen „########" Zeichen.
Was nutzt mir das schönste Codewort, wenn es jeder auf dem Bildschirm lesen kann, den es nichts angeht.
Übrigens, das Codewort für das abgedruckte Programm ist schlicht und einfach START! Wenn Sie das Wort ändern wollen in Zeile 150, beachten Sie bitte in Zeile 160: Len‹T$›=6 ......, sobald jemand den sechsten Buchstaben eingibt, geht's ab nach Zeile 300, wo der Computer ganz deutlich zu verstehen gibt: # ZUGRIFF VERWEIGERT #
Also in Zeile 160 immer 1 Buchstabe mehr angeben, als das Codewort hat.

```
75 REM ***************
80 REM * 'CODEWORT!' *
85 REM ***************
90 T$="":PRINTCHR$(147)
100 PRINT"CODEWORT BITTE":PRINT:PRINT:PRINT
120 GETA$:IFA$=""THEN120
140 T$=T$+A$
150 IFT$="START"THEN200
160 IFLEN(T$)=6THEN300
165 PRINT"#";
170 GOTO120
200 PRINTCHR$(147):PRINT"# ZUGRIFF #"
230 END
300 PRINTCHR$(147):PRINT"# KEIN  ZUGRIFF #"
310 FORT=1TO2000:NEXT
320 GOTO90

READY.
```

Datum-Input-Kontrolle

BITTE DATUM EINGEBEN: So spricht der Computer per Bildschirm zu mir. Jetzt kommt bei mir die große Frage auf: Wie hätte Herr Computer es denn gerne?

- mit Nullen-Eingabe bei Tag und Monat?
- keine Trennzeichen bei Tag und Monat?
- ist der 31. April oder der 31. Juni akzeptabel?
- wie sieht's aus beim 29. Februar 1985?
- wird die Meldung „REDO FROM START" erscheinen?

Davon wird in den wenigsten Programmen etwas erwähnt. Vielleicht liegt die Ursache dieses Problems bereits in der Wurzel des Programms; nämlich beim Programmierer!
Ein Programmierer sollte eigentlich die Datum-Input-Routine so gestalten, daß der Anwender keine Fehleingaben machen kann. Sollte es doch einmal dazu kommen, dürfte der Bildschirmaufbau nicht durch die REDO FROM START-Meldung verunstaltet werden! Dies alles läßt sich doch 100%ig zufriedenstellen, wenn der Computer alle Eingaben überprüft und kontrolliert.
Ich denke, dafür gibt es eine kleine Routine, die sich an jeder Stelle eines Programmes einbauen läßt, ohne viel Platz zu benötigen.
Tippen Sie die paar Zeilen einmal ab und machen den Test auf „Herz und Nieren"! Es wird Ihnen sicherlich viel Spaß und Freude bereiten.
Dieses Basic-Programm wurde auf COMMODORE 64 und VC-20 geschrieben. Es dürfte jedoch auch auf anderen Computern ohne Veränderungen laufen. *R. Petruck*

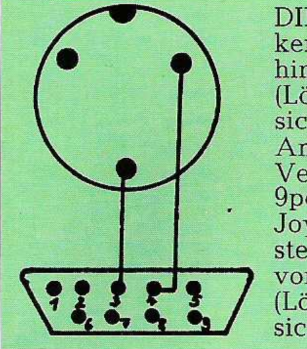
```
10 :
20 REM ************************
30 REM * DATUM-INPUT-CONTROLL *
40 REM ************************
50 PRINTCHR$(147)
60 PRINT"DATUM EINGEBEN"
65 PRINT
70 PRINT"FORMAT: TTMMJJ"
80 PRINT:PRINT
90 PRINT:PRINT
100 INPUT"DATUM: ";DAT$
110 L=LEN(DAT$)
120 IFL<>6THEN 50
130 TT$=LEFT$(DAT$,2)
140 MM$=MID$(DAT$,3,2)
150 JJ$=RIGHT$(DAT$,2)
200 TT=VAL(TT$):MM=VAL(MM$):JJ=VAL(JJ$)
210 IF TT <1 OR TT >32 THEN 50
220 IF MM <1 OR MM >13 THEN 50
230 IF JJ <1 OR JJ >99 THEN 50
250 TEST$="312931303130313130313031"
260 AT$=MID$(TEST$,MM+MM-1,2)
280 IFTT>VAL(AT$)THEN 50
290 IFMM <>2 THEN 500
300 AJ=INT(JJ/4)
310 IFAJ<>JJ/4 AND TT> 28 THEN 50
500 PRINTCHR$(147)
510 PRINT"DATUM OK !"

READY.
```

Konzertreife Stimme für den TI 99/4A

Wer den Sound seines TI 99 aufmotzen will, braucht dazu nur einen Radiorecorder und die nachfolgende Bauanleitung.

Wer sich mit dem flachen, dumpfen Ton eines 198 DM-Fernsehers herumschlagen muß, ist sicher dankbar für eine kleine Verbesserung der Tonmöglichkeiten.
Daß kleine Qualitätssteigerungen nicht unbedingt teuer und aufwendig sein müssen, beweist diese Bauanleitung.
Die Voraussetzungen dafür sind denkbar gut: An der Interfacebuchse für das Kassettenrecorderkabel liegt ein Audiosignal an, das sich zum Ansteuern eines Radiorecorders, eines einzelnen Verstärkers oder sogar einer Stereoanlage eignet. Dazu benötigt man lediglich einen 9poligen Joystickstecker, mit dem man den Tonausgang des Computers anzapft und einen passenden Stecker für den Anschluß an den Verstärker. Hier findet ein dreipoliger DIN-Stecker bei den meisten Radiogeräten Verwendung. Bei fast allen Radiorecordern reicht nach meiner Erfahrung eine einfache Verbindung zwischen Computer und Audioeingangsbuchse des betreffenden Gerätes aus. Will man einen HiFi-Verstärker oder ein ähnliches großes Gerät anschließen, ist jedoch fast immer noch ein Vorverstärker nötig, der zwischen Computer und Stereoanlage geschaltet wird.



DIN-Stekker von hinten (Lötfahnen sichtbar) für Anschluß an Verstärker 9poliger Joystickstecker von hinten (Lötfahnen sichtbar)

Die genaue Verdrahtung können Sie der Abbildung entnehmen. Der Tonausgang des TI 99 kann an der 9poligen Buchse an der Geräterückwand mit Pin 4 gegen Masse abgegriffen werden. Die Verbindung mit dem externen Verstärker stellt ein zweiadriges, nicht unbedingt abgeschirmtes NF-Kabel dar. Bei einem längeren Kabel empfiehlt sich trotzdem eine Abschirmung, um einen Netzbrummen zu vermeiden.
Nach der Verdrahtung sollte man unbedingt noch einmal alles auf seine Richtigkeit überprüfen, da Fehler sowohl Computer als auch den angeschlossenen Verstärker beschädigen könnten.
Will man den Tonausgang des TI 99 für die Ansteuerung eines externen Verstärkers benutzen, muß man die Recorderkabelbuchse jedesmal dafür freimachen. Deshalb für die User noch einen Tip, die sich auch das Recorderkabel selbst zusammengelötet haben: Der Verstärkeranschluß kann natürlich auch mit in dieses Kabel eingelötet werden. Dies ist bei gekauften Kabeln leider nicht möglich, da die Stekker fest verschweißt sind.
Noch viel Spaß beim Zusammenbau und bei den zukünftigen Videospielen, bei denen es ab jetzt kristallklar und hautnah kracht!

Stückliste:
1 Joystickstecker
1 Stecker, passend für benutzten Verstärker,
evt. Vorverstärker für Stereoanlagen

Impressum: Computronic erscheint monatlich im Verlag: Tronic-Verlagsgesellschaft mbH, Landstr. 29, 3444 Wehretal 1, Tel.: (0 56 51) 4 06 43 / 4 06 93. Redaktion: (Verantw.) Rainer Beck, Axel Gredè, Frank Brall. Freie Mitarbeiter: G. Wagner, Volker Bekker, Rolf Freitag. Gesamtherstellung: D + V Paul Dierichs GmbH + Co. KG. Kassel. Vertrieb: Inland (Groß-, Einzel- und Bahnhofsbuchhandel), sowie Österreich und Schweiz: Verlagsunion, 6200 Wiesbaden. Anzeigenverwaltung: E. Herwig. Anzeigenpreis: Es gilt die Anzeigenliste Nr. 1. Bitte Media-Unterlagen anfordern. Bezugspreise: Einzelheft (Inland) DM 5,50, Abonnement (Inland) DM 55,-, (Ausland) DM 75,-. Autoren und Manuskripte: Bei Zusendung von Manuskripten und Datenträger erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger. Für die mit Namen des Verfassers gekennzeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung.

Computer-Börse

Verkäufe

Der ganz heiße Börsentip vom eigenen Computer
Stockmaster: grafische **Stock-Control:** mathemat. Auswertung f. Spectrum 48K je Pr. 49,- DM zus. nur 79,- DM.
Töngi
Aspeltstraße 4
6500 Mainz

Verkaufe das Buch Atari Basic Hofacker Nr. 32 für 25 DM, 1 Monat alt. Außerdem verkaufe ich Atari Software (Zaxxon ...). Liste gegen 80 Pf in Briefmarken bei Mario Noack, Raabestr. 36, 3420 Herzberg

Verk. **ZX 81/16 KCMEMO** + 80 Pr. + Reset + Shift Look + Rem-Ansteuerung + gr. Tastatur + Bücher + 90 Listings + Ein/Aus + gr. Netzteil für 350,- DM: Suche Spektrum 48 K, T. Bachmann, Johannisbruch 9, 2357 Springe 1

TI 99/4A + Exbasic + div. Bücher + Rec. + Kabel + Basic-Kurs, VB 750,- DM.
Tel.: 0 28 61 / 48 45
Sa. ab 18.00 Uhr

Verkaufe **ZX-81 + 16 K + Drucker** VB 300,- DM. Suche Atari- 600XL oder C-64. Dirk Wylezol, Am Mühlbachbogen 41 b, 8052 Moosburg

C-64-Superprogramme gegen Unkosten oder Tausch. Gratisliste bei Postlagernd 058542 B, 7900 Ulm/Donau, Diskette oder Kassette

Atari 400/800/600 XL/800 XL Software auf: C. + D. Liste anfordern bei: Power-Game Versand, Oerlinghauser Str. 37 b, 4800 Bielefeld 18

Verkaufe
Anrufbeantworter
Drahtlostelefon
Telefon-Anlage
Eurosignal 1200,– DM
div. USA-Telefone
Info bei:
Reinhard Winner
Höchbergerstraße 62
87 Würzburg

Software für Commodore, Tel. 02 74 72 99

Wichtige Toolkits für Spectrum & C64, Spectrum Disassembler 1.1 (48 k), C64 Sprite Construction Set ZX81 OMELS, ein Strategiespiel (16 k). Diese Pgms. u.v.a.m. für je DM 20,– bei Daniel Schwinn, Meisenweg 6, 7073 Lorch

Verkaufe **Taschenrechner TI 59,** Disketten 8 Zoll SS/SD, Preise VHS. Willi Hirschlein, H. v. Hohenlohestr. 44, 6990 Bad Mergentheim, Tel. 0 79 31 / 4 16 05

TI-99/4A+EXT Basic: universelle Datenverwaltung mit Sortierfunktion u. 2 Suchalgorithm., 15.– DM inkl. Cass. + Anleitung, Geld/Scheck an Dirk Storck, Weserstr. 23, 4840 Rheda-WD

ZX 81 mit 64 KB, 5 Monate alt, in Philips-Terminal zu verkaufen, als Beigabe 15 Spiele für den ZX 81. Angebote an: M. Kraemer, Weidenweg 6, 4970 Bad Oeynhausen.

Here is Wimpy!
Hello User, biete für C-64 Disk beidseitig mit Topprogrammen für DM 50,- auch Tausch.
Mr. Wimpy
4350 Recklinghausen
Postfach 1017 21

Commodore C 64 nur DM 679,-, Floppy 1541 nur 679,-, Drucker MPS 801 nur DM 669,-. Disketten 10 Stck. 55,-. Alle Geräte neu mit Garantie. Anruf lohnt sich!!! Tel.: 0 75 62 / 34 30!!!

TI 99/4 A Ext.-Spiel nur für Erwachsene – bitte Alter ang. zum absoluten Preishit von nur DM 10,- im Umschl. an T. Karbach, Remscheider Str. 18, 5650 Solingen 1

Wenn ihr „Mugsy von Melbourne House" wollt, dann kriegt ihr es für 35 DM (vielleicht) noch bei: Jan Weigner, Carl-Schurz-Str. 11, 2800 Bremen 1, Tel. 04 21 / 34 63 87

TI 99/4A. Erstelle Druckerlisting v. Ihren Programmen in TI- u. ext. Basic, je Pgm.-Zeile 4 Pf. + 2,50 DM f. Portokass. m. V.-Scheck o. Briefmarken an H. Schwab, Loch 21, 8504 Stein

Ti-99/4a und alle Atari-Computer für Ti:
Adventure Modul mit Cassette Mission Impossible und Pirate Adventure: 160,- DM für Atari Star Raiders für 70,- DM bei
O. Toffolo
Tel. 0 46 62 / 17 33

★★★ FLOPPY 1541 6 x SCHNELLER ! ★★★ durch Parallelbus für C 64. Interface kpl. geprüft mit Software nur DM 179,–. Info gegen DM 1,50 in Briefmark., B. Akesson, Pf. 802, 4040 Neuss.

Ankäufe

TI-99/4A
Suche **Extended Basic Modul** evtl. Tausch gegen **Editoriassembler** sowie Maschinensprachprogramme z. B. **Flugsimulator** und **Ex. Basic Spiel Miner 2049** (USA-Version).
Tel. 0 71 32 / 3 76 08

Apple II suche Programme, Schachspiele, Flug-Simulation, versch. Vorspanntitel, Grafik für Video-Filme, T. Pavone, Heilbronnstr. 6, 7146 Tamm

Suche **Seikosha 6P-100A** Drucker mit Interface für ZX Spectrum, Lothar Schröder, Dellchenstraße 3, 6731 Frankeneck

Suche: Gute und billige Software auf Diskette für 64K Atari 600XL. Angebote an: Michael Ebeling, Riepener Straße 7, 3061 Beckedorf

Suche **Atari Diskettenstation,** Aussehen egal, Hauptsache sie funktioniert. Bitte anrufen unter Berlin 0 30 / 4 93 – 19 27

ZX Spectrum Soft- und Hardware. Preiswert durch Direktimport. Info bei: Ursula Kunz, Junge Hälden 3, 7500 Karlsruhe 41

Suche **Software** für Atari 600XL, zahle gut! Schickt eure Liste an: Mario Noack, Raabestr. 36, 3420 Herzberg

Kontakte

Hallo Spektrumfreaks. Ich möchte mit euch gerne Prog. tauschen. Liste, ob groß oder klein, an: Robert Weidinger, Schönmetzlerstr. 10, 8050 Freising oder Sams.– Diens. 0 81 61 / 43 72

Kontakt mit MZ700-Leuten gesucht. Tel. 05 11 / 42 29 75

***** Atari-Spectrum-User-Club *****
monatl. Info, bundesweit organisiert, riesige Prog.-Bibliothek usw., melden bei:
ASUC, Am Schürhof 17
4905 Spenge
Tel. 0 52 25 / 20 68

Comp.-Club
Bad Hersfeld
sucht Mitglieder!
Kontakt über
Roland Reyer
Am Giegenberg 21
6438 Ludwigsau 1
oder Jugendhaus Hersfeld

Tausch

Tausche **ZX Spectrum-Software** (16/48), auch Microdrive-Software benötigt.
Michael Jedral
Horststraße 6
6729 Leimersheim.
Schreibt schnell. Suche Kopierprogramme für Microdrives

C-64-Superprogramme gegen Unkosten oder Tausch. Gratisliste, Postlagernd 05 85 42 B, 7900 Ulm/Do. Nur neueste Prg.

VC 64: Tausche irre Superprogramme, über 40 echte Spielhallen-Games. Liste gegen Rückporto bei Michael Brandenberger, Vor der Gasse 82, CH-8200 Schaffhausen

ZX-Spectrum
Software und Interessenaustausch
Monika Baumgartner
4021 Linz
Postfach 142 Austria
Bitte Liste beilegen

Verschiedenes

16 K Modul f. ZX 81, 65,- DM. Trak-Ball für Atari + Commodore, 1 Fevertaste, 99,- DM. Beides neu, unbenutzt. Verkaufe (2 DM – 4 DM) + tausche Spectr. Softw. Liste gr. T. Schröer, Hohlstr. 11, 6791 Steinbach

UFO-Forschung. Informationen über UFOs etc. kostenlos. Suche Interessenten zum Aufbau einer Computer-AG (c 64), spez. für wiss. UFO-Forschung! H. W. Peiniger, PF 2361, 5880 Lüdenscheid

VZ 200 Softw. Info g. Freiumschlag. W. Fischer, Friedrichstr. 14, 6463 Freigericht 5

Orig. engl. Software von EMJAY, (Wargame News, The War Machine), für Commodore-Rechner ab 16k und CBM 64! Näheres gegen Freiumschlag von: H. Topf, A.-Bucherer-Straße 63, 5300 Bonn 1

Die richtige Soft-Hardware? Ich sage Ihnen in welcher Zeitschrift. Info gegen frank. Rückumschlag bei: M. Kiwitt, 2000 Hamburg 74, Postlage RND

TI 99/4A Software Service von Action bis Praxis. Info geg. 1. – Rückporto Behringstr. 45, 4600 Dortmund 50

C-64-Synthesizer: ca. 40 K, echte Noten, Partiturediting, Notenausdruck, Direkteinspielung + Notenumsetzung, Supersound. Info von Autor W. Kracht, Espellohweg 38, 2000 Hamburg 52

Roulettesoftware f. Sharp PC 1211/12/4551/PC 1500. Roulettesystemprogramme »Auf Dauer gew.« »AD Astra« »Number one« «Acclaim« »Plein Prive« »Accord« DM 550,- Cass. B. Nowotny, Giselastr. 3, 8000 München 40

An alle SF- und Fantasy-Freunde! Jeweils 8 Mitspieler für Strategiespiele gesucht. Mit Gewinnchancen. Näheres gegen Freiumschlag von: H. Topf, A.-Bucherer-Straße 63, 5300 Bonn 1

ZX-SPECTRUM! Spielend BASIC lernen Ca. mit 10 Pr.+Listing+Erklärung+Tips+Tricks für den ZX. 10,- DM bei A. Meuser, Goethestr. 18, 5205 St. Augustin NEU - SUPER ZX Spiele -

Spectrum Softw. z. B. M. Miner 8,-; Z00M 8,-; Liste + kostl. Programm anfordern, Karte genügt: H. Novak, Egerländer Pl. 19, 6053 Oberts-hausen 2

TI-99/4A: ca. 90 TI-Basic und ca. 80 Ex-Basic Programme zu verk. Info g. Rückumschlag. B. Knedel, Tulpengasse 16, 3171 Weyhausen, Tel. 0 53 62 / 7 11 87



Aktion: Billige Kleinanzeigen

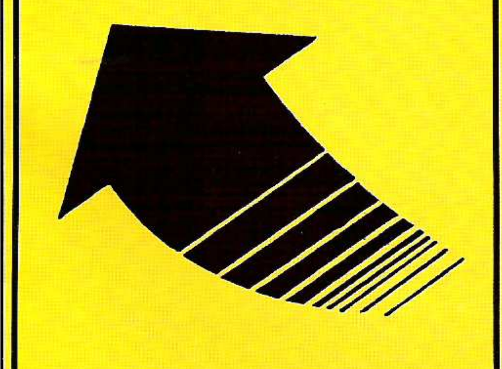
Liebe Leser,
wollen sie ein Kleinanzeige aufgeben? Suchen Sie einen gebrauchten Home-Computer, Software oder Kontakte zu Computer-Clubs etc.?

Dann nutzen Sie jetzt den Kleinanzeigenservice von „Computronic". „Computronic" ist die Zeitschrift für Home-Computer-Besitzer. Sie erreichen durch uns eine Vielzahl von Lesern im In- und Ausland.

Und so wird's gemacht:
Kleinanzeigen-Coupon ausfüllen (alle Zeilen können ausgefüllt werden) und gegen
Vorkasse von DM 4,50 einsenden an:

Tronic-Verlag
Postfach 41
3444 Wehretal 1

Also, schnell ausfüllen und einwerfen in den nächsten Postkasten. Dann ist mit Sicherheit Ihre Kleinanzeige dabei.



Computronic Kleinanzeigen nur gegen Vorkasse

Bitte veröffentlichen Sie ab nächstmöglicher Ausgabe unter der Rubrik:

☐ Verkäufe ☐ Kontakte ☐ Geschäftsempfehlungen
☐ Ankäufe ☐ Verschiedenes ☐ Tausch

folgenden Text:

Bitte jeweils 28 Buchstaben pro Zeile! Bitte Absender nicht vergessen!

Preis für Kleinanzeigen:		
Private Anzeige	DM 4,50	je Zeile
gewerbl. Anzeige	DM 8,00	je Zeile
Chiffregebühr	DM 10,00	je Anzeige

Name/Vorname:

Straße, Nr.: PLZ/Ort:

Datum, Unterschrift

privat ☐ gewerbl. ☐
(bitte ankreuzen)

Kassettenservice

HEFT 4

VC-64 K = 8,--DM
D = 15,--DM
Mauern, Widerstand

ZX-Spectrum K = 12,--DM
Inventur

TI 99 K = 8,50DM
Drei-Kronen-Spiel
Zahlenputzen

VC-20 K = 11,50DM
D = 18,--DM
Hangman, Saurer Regen, Quadr. Gleichung

ZX-81 K = 8,50DM
Invader, Gun-Man

Dragon 32 K = 8,--DM
Blizzard

Apple II K = 14,50DM
D = 19,50DM
Wilder Westen, Karambolage, Maskengenerator

Atari K = 10,50DM
Mastermind, Schlangenkrieg

HEFT 5

TI 99 K = 14,50DM
Karl der Käfer
Alien-Landing

VC-64 K = 15,50DM
D = 23,50DM
Space-Comets/Erdspalte/Sprite-Data

Apple II K = 9,50DM
D = 19,50DM
Musik-Maker/Mission-Adler/Disk-Catalog

Sharp MZ 700 K = 8,50DM
Kalender
Sharp PC 1500 Lotto

Dragon 32 K = 10,--DM
Space-Flight, Geosoft

ZX-81 K = 10,--DM
Go-Ball, Grand-Prix

ZX Spectrum K = 8,50DM
Missile-Comment

Atari K = 12,50DM
Tank-Battle/Oil Panic

HEFT 6

VC-64 K = 16,50DM
D = 23,50DM
Autostart/Bestellschein/Roadpainter

Dragon 32/64 K = 8,50DM
Wargames

Apple II K = 12,50DM
D = 19,50DM
Snake/Super Datei/Shapetables

VC-20 K = 8,50DM
D = 15,00DM
Bestellschein/Glücky

ZX-81 K = 10,--DM
Moon-Crash/ZX-Draw

ZX-Spectrum K = 13,50DM
Defender/Lui der Wurm/Alternativer Zeichensatz

TI-99 K = 14,50DM
Jack the Digger/Noah – 2099

Atari K = 12,50DM
Tomstone-City/Schneevogel

HEFT 7

VC-64 K = 15,50DM
D = 19,50DM
Hardcopy/Space-Fighter/Data-Generator

ZX-81 K = 10,--DM
Tonprogramm/Aldebaran

Atari K = 12,50DM
Startup/Zeilen-Split/Chopper-Flight

VC-20 K = 11,--DM
D = 15,50DM
Multigraph/All-Rammer

ZX-Spectrum K = 12,50DM
Matheprogramm/Bongo-Beecatcher

Apple II K = 12,50DM
D = 19,50DM
Library/Fight

Dragon 32 K = 8,50DM
Laser-Attack

TI-99 K = 14,--DM
Lift Bär
ASC II DEF TEIL 1



Computronic Bestellkarte-Kassettenservice

Alle im Heft abgedruckten Programme können als zusätzlicher Service, zum Selbstkostenpreis, über den Verlag bezogen werden. **(Ausland nur gegen Vorkasse)**

Die Zustellung erfolgt: gegen **Vorkasse** ☐ oder Inland per **Nachnahme** ☐ + Versandkosten

innerhalb von 1 Woche

Entnehmen Sie bitte aus unserer Preisliste die notwendigen Angaben für Ihre Bestellung:

Bitte liefern Sie mir:

☐ Cassette für ____________ System ☐ Anzahl

☐ Diskette für ____________ System ☐ Anzahl

zum Preis von gesamt ____________ DM

Name/Vorname: ____________ Straße, Nr.: ____________

PLZ/Ort: ____________ Datum, Unterschrift ____________

Kassettenservice

Jedes Programm in Computronic wird einer Endkontrolle in unserem Hause unterzogen und genauestens geprüft. Alle im Heft abgedruckten Programme sind nach der richtigen Eingabe der Listings auch funktionsfähig. Viele Leser verlieren jedoch schnell die Geduld am Programmieren, sollte etwas einmal nicht klappen. Die häufigste Ursache von Störungen im Programm, sind unterlaufene Fehler bei der Eingabe. Verzweifeln Sie nicht, sollten Sie einmal keine Zeit zum Programmieren haben oder sollte etwas nicht gelingen. Alle Programme werden im Verlag gespeichert und können jederzeit mit beiliegender Bestellkarte zum Selbstkostenpreis bezogen werden. Tragen Sie bitte alle notwendigen Angaben in die Bestellkarte ein.

Rückgabe-Garantie:
Wir garantieren:

- kostenlosen Umtausch von defekten bzw. transportgeschädigten Datenträgern!
- die Zusendung der Umtauschware erfolgt **noch am Posteingangstag!**

Bestellung per Telefon:
Wenn es schnell gehen soll ... rufen Sie uns an. Wir nehmen Ihre Bestellung gern entgegen.
Tel.-Nr.: 0 56 51 – 4 06 93 oder **0 56 51 – 4 06 43**
Tronic-Verlag, Postfach 41, 3444 Wehretal 1

Bitte beachten Sie:
Sie ersparen sich zusätzliche Kosten (bis zu DM 5,–), wenn Sie per Vorkasse (bar, Verrechnungsscheck) bestellen.
Ausland: ● Bestellung nur gegen Vorkasse!

Tontaubenschießen C-64 K = 16,00 DM D = 19,50 DM
Ein Spiel mit sehr guter Grafik für Ihren C-64. Nach den Regeln des sportlichen Tontaubenschießens müssen Sie versuchen, alle 25 Tauben, die einzeln oder auch als Dublette vom kleinen bzw. großen Haus kommen, zu treffen.

HEFT 8

System	Preis	Programme
C-64	K = 16,50 DM D = 23,50 DM	Monster Attack/Block-Painter/Epson-Drucker
Atari	K = 14,-- DM D = 19,50 DM	Painter/Hardcopy
Apple II	K = 14,-- DM D = 19,50 DM	Reversal Disk-Menue-Generator
TI-99	K = 14,50 DM D = 19,50 DM	Maya/ASC II DEF Teil 2
ASC II DEF Teil 1+2	D = 19,50 DM D = 21,-- DM	
Laser 2001	K = 8,50 DM	Andromeda
ZX-81	K = 10,-- DM	Irrgarten 3 D
ZX-Spectrum	K = 14,50 DM	Solitaire/Superstat. Kleinstes gem. Vielfaches
VC-20	K = 11,-- DM D = 19,50 DM	Zyklo/Meteorit
Dragon	K = 13,-- DM	Hardcopy

HEFT 9

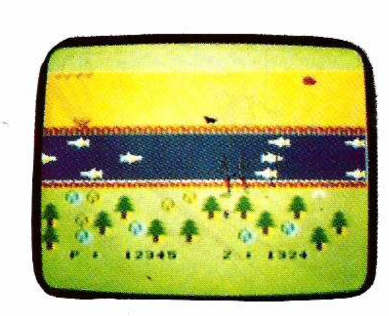
System	Preis	Programme
Colour Genie	K = 10,-- DM	Fuchs und Hund
C-64	K = 16,-- DM D = 23,50 DM	Projekt Datenbank
Atari	K = 14,-- DM D = 19,50 DM	The Big Quest Fünf gewinnt
Apple	K = 14,-- DM D = 19,50 DM	Diamonds Hilfsprogramm
TI 99	K = 14,50 DM D = 19,50 DM	Transfer Silverspar
Laser 2001	K = 12,-- DM	Crazy Cake
ZX-81	K = 10,-- DM	Reversi
ZX-Spectrum	K = 14,50 DM	Jump about
VC-20	K = 14,-- DM D = 19,50 DM	Garten Schloß Gruselstein
Dragon	K = 10,-- DM	Anwenderprogramm

Im nächsten Heft

wieder
Neues, Infos, Tips & Tricks
unser
Forth-Kurs
und natürlich wieder tolle
Spiel- und **Anwenderprogramme.**

Für den *TI-99* bringen wir u. a.:

Mother Duck



Sind Sie schon einmal in die Rolle einer Entenmutter geschlüpft, die verzweifelt versucht, ihre Kinder zu füttern?

Screen Designer
Ein Maskengenerator, in Extented-Basic geschrieben.

Für den *Laser 2001:*
Cave Man
Cave Man zeichnet sich besonders durch die hervorragende Grafik aus. Leo Neandertal lebt in der Steinzeit. Um sich und seine Familie ernähren zu können, muß er täglich auf die Jagd gehen. Am einfachsten ist es aber, dem Saurier Rex die Eier zu stehlen.

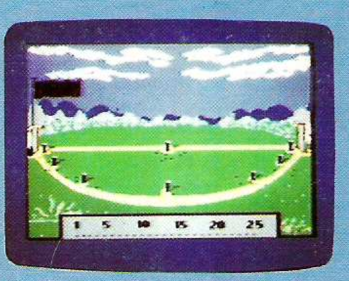
Natürlich berücksichtigen wir wieder alle gängigen Homecomputer. Systeme wie: *Commodore 64, VC-20, ZX-Spectrum, ZX-81, Apple II, Dragon 32* und den *Atari.*

Also, auf zum Zeitschriftenhändler, *ab 24. 9. 84* gibt es wieder eine neue „Computronic"-Ausgabe.

Es verabschiedet sich:
Die Computronic-Redaktion

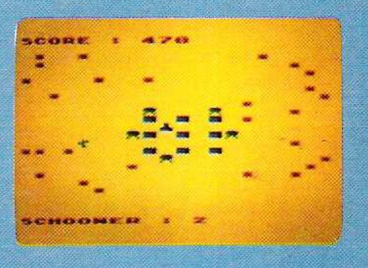
Kassettenservice aus unserem Angebot Seite 78 und 79:

Tontaubenschießen:



Für **Commodore 64.** Dem realistischen Tontaubenschießen nachempfunden. Von einem Katapult geschleuderte Tontauben müssen reaktionsschnell getroffen werden. In diesem Spiel kann jeder seine eigene Meisterschaft austragen. Eine Supergrafik zeichnet dieses „Tontaubenschießen" besonders aus. In Computronic wurde dieses Programm noch nicht ausgedruckt.

Tomstone-City:



Eine kleine Stadt im „Wilden Westen" gewährt Dir Schutz vor Deinen Feinden. Solange Du sie nicht verläßt kann Dir nichts passieren. Außerhalb der Stadt wirst Du jedoch gejagt. Gewählt werden kann zwischen verschiedenen Schwierigkeitsgraden. Ein Spiel für **Atari 600/800 XL.**

Projekt:

Als Topprogramm auserwählt von der Redaktion. Sie sollen eine gefährliche Mission ausführen. Um für die nächste Zeit genügend Uran zur Verfügung zu haben, sollen Sie eine Reise zum Mond unternehmen und nach dem edlen Metall suchen. Ihre Reise ist in fünf Phasen aufgegliedert, die nacheinander bewältigt werden müssen. Ein schönes Spiel für den **Commodore 64.**

Maya:



Das Spiel läuft auf dem **TI-99/4 A** mit dem **Extented Basic Modul** und wird mittels Joystick gesteuert. Der Rechner kann in diesem Spiel bis zu 378!! verschiedene Bildschirmszenen darstellen. Das Spiel verfügt über eine hervorragende grafische Darstellung. Begleiten Sie also Coconut Joe auf die abenteuerliche Expedition in den Urwald. Vielleicht haben Sie Glück und finden die Schätze der Mayas.

Monster-Attak:



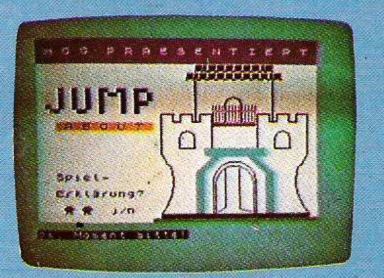

Für den **Commodore 64.** Fremde Wesen greifen die Erde an. Versuchen Sie die angreifenden Monster vor Erreichen der Erde zu zerstören. Das Spiel unterteilt sich in 6 Level, wobei bei jedem Level die Spielgeschwindigkeit sowie die erreichbare Punktzahl ansteigt. Das Spiel wird mit Joystick gespielt.

Crazy-Cake:



Backe, backe Kuchen ... Setzen Sie Ihre Bäkkermütze auf, denn hier kommt ... die verrückte Kuchenbäckerei.
Das Spiel ist in Laserbais für den **Laser 2001** geschrieben und wird mit Joystick gesteuert.

Jump about:



Ein kleiner Floh und sein Freund haben ein schweres Schicksal zu meistern. Sie sind beim Herumhüpfen unbewußt in den Einflußbereich eines bösen Magiers geraten. Nun gilt es, den Weg in die Freiheit zu finden. Ein Spiel für den **ZX-Spectrum** mit toller Graphik.

Computronic erscheint monatlich im Tronic-Verlag. Der Bezugspreis beträgt im Inland DM 5,50.
Im Preis ist die gesetzliche Mehrwertsteuer enthalten.
Tronic-Verlag GmbH · Postfach 41 · 3444 Wehretal 1 · Telefon 0 56 51 - 4 06 43 / 4 06 93